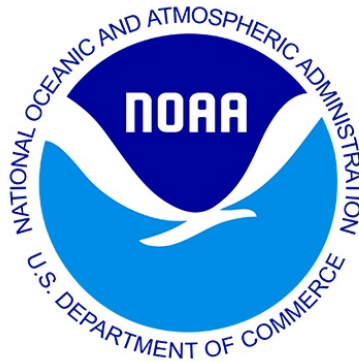# Stock Synthesis User Manual
Version 3.30.22.1

Richard D. Methot Jr., Chantel R. Wetzel, Ian G. Taylor, Kathryn L. Doering, Elizabeth F. Perl, and Kelli F. Johnson

NOAA Fisheries
Seattle, WA

January 31, 2024

Contents

Introduction

Fish population (aka "stock") assessment models determine the impact of past fishing on the historical and current abundance of the population, evaluate sustainable rates of removals (catch), and project future levels of catch reflecting one or more risk-averse catch rules. These catch rules are codified in regional Fishery Management Plans according to requirements of the Sustainable Fisheries Act. In the U.S., approximately 500 federally managed fish and shellfish populations are managed under approximately 50 Fishery Management Plans. About 200 of these populations are assessed each year, based on a prioritized schedule for their current status. Despite this, many minor species have never been quantitatively assessed. Although the pace is slower than that for weather forecasting, fish stock assessments are operational models for fisheries management.

Assessment models typically assimilate annual catches, data on fish abundance from diverse surveys and fishery sources, and biological information regarding fish body size and proportions at age. A suite of models is available depending on the degree of data availability and unique characteristics of the fish population or its fishery. Where feasible, environmental time series are used as indicators of changes in population or observation processes, especially to improve the accuracy of the projections of abundance and sustainable catch into the future. Such linkages are based principally on correlations given the challenge of conducting field observations on an appropriate scale. The frontier of model development is in the rapid estimation of parameters to include random temporal effects, in the simultaneous modeling of a suite of interacting species, and in the explicit treatment of the spatial distribution of the population.

Assessment models are loosely coupled to other models. For example, an ocean-temperature or circulation model or benthic-habitat map may be directly included in the pre-processing of the fish abundance survey. A time series of a derived ocean factor, like the North Atlantic Oscillation, can be included as an indicator of a change in a population process. Output of a multi-decadal time series of derived fish abundance can be an input to ecosystem and economic models to better understand cumulative impacts and benefits.

Stock Synthesis is an age- and size-structured assessment model in the class of models termed integrated analysis models. Stock Synthesis has evolved since its initial inception in order to model a wide range of fish populations and dynamics. The most recent major revision to Stock Synthesis occurred in 2016, when v.3.30 was introduced. This new version of Stock Synthesis required major revisions to the input files relative to earlier versions (see the Converting Files section for more information). The acronym for Stock Synthesis has evolved over time with earlier versions being referred to as SS2 (Stock Synthesis v.2.xx) and older versions as SS3 (Stock Synthesis v.3.xx).

SS3 has a population sub-model that simulates a stock's growth, maturity, fecundity, recruitment, movement, and mortality processes, an observation sub-model estimates expected values for various types of data, a statistical sub-model characterizes the data's goodness of fit and obtains best-fitting parameters with associated variance, and a forecast sub-model projects needed management quantities. SS3 outputs the quantities, with confidence intervals, needed to implement risk-averse fishery control rules. The model is coded in C++ with parameter estimation enabled by automatic differentiation (admb). Windows, Linux, and iOS versions are available. Output processing and associated tools are in R, and a graphical interface is in QT. SS3 executables and support material is available on GitHub. The rich feature set in SS3 allows it to be configured for a wide range of situations. SS3 has become the basis for a large fraction of U.S. assessments and many other assessments around the world.

This manual provides a guide for using SS3. The guide contains a description of the input and output files and usage instructions. An overview and technical description of the model itself is in Methot and Wetzel (2013). However, SS3 has continued to evolve and grow since the publication in 2013, with this manual reflecting the most up to date information regarding SS3. The model and a graphical user interface are available on GitHub with older archived versions also available online at NOAA VLAB. The VLAB site also provides a user forum for posting questions and for accessing various additional materials. An output processor package, r4ss, in R is available for download from CRAN or GitHub.

Additional guidance for new users can be found online in the Getting Started Tutorial and on the Stock Synthesis GitHub page.

How To Cite

Please cite Stock Synthesis as:

Methot, R.D. and Wetzel, C.R. (2013). Stock Synthesis: A biological and statistical framework for fish stock assessment and fishery management. Fisheries Research, 142: 86-99. https://doi.org/10.1016/j.fishres.2012.10.012

Please cite the Stock Synthesis User Manual as:

Methot, R. D., Jr., C. R. Wetzel, I. G. Taylor, and K. Doering. (2020). Stock Synthesis User Manual Version 3.30.15. U.S. Department of Commerce, NOAA Processed Report NMFS-NWFSC-PR-2020-05.https://doi.org/10.25923/5wpn-qt71

File Organization

Input Files

1. starter.ss: required file containing filenames of the data file and the control file plus other run controls (required).

2. datafile: file containing model dimensions and the data (required)

3. control file: file containing set-up for the parameters (required)

4. forecast.ss: file containing specifications for reference points and forecasts (required)

5. ss3.par (previously ss.par): previously created parameter file that can be read to overwrite the initial parameter values in the control file (optional)

6. wtatage.ss: file containing empirical input of body weight by fleet and population and empirical fecundity-at-age (optional)

7. runnumber.ss: file containing a single number used as run number in output to CumReport.sso and in the processing of profilevalues.ss (optional)

8. profilevalues.ss: file contain special conditions for batch file processing (optional)

Output Files

1. data_echo.ss_new: Contains the input data as read by the model. In model versions prior to v.3.30.19 a single data.ss_new file was created that included the echoed data, the expected data values (data_expval.ss), and any bootstrap data files selected (data_boot_x.ss).

2. data_expval.ss: Contains the expected data values given the model fit. This file is only created if the value for "Number of datafiles to produce" in the starter file is set to 2 or greater.

3. data_boot_x.ss: A new data file filled with bootstrap data based on the original input data and variances. This file is only created if the value in the "Number of datafiles to produc" in the starter file is set to 3 or greater. A separate bootstrap data file will be written for the number of bootstrap data file requests where x in the file name indicates the bootstrap simulation number (e.g., data_boot_001.ss, data_boot_002.ss,...).

4. control.ss_ new: Updated version of the control file with final parameter values replacing the initial parameter values.

5. starter.ss_ new: New version of the starter file with annotations.

6. Forecast.ss_ new: New version of the forecast file with annotations.

7. warning.sso: This file contains a list of warnings generated during program execution. Starting in v.3.30.20 warnings are categorized into either Note or Warning. An item marked as a not denotes settings that the user may want to revise but do not require any additional changes for the model to run. Items marked with Warning are items that may or may not have allowed the model to finish running. Items with a fatal warning caused the model to fail during either reading input files or calculations. Warnings classified as error or adjustment may be causing calculation issues, even if the model was able to finish reading file and running, and should be addressed the user.

8. echoinput.sso: This file is produced while reading the input files and includes an annotated echo of the input. The sole purpose of this output file is debugging input errors.

9. Report.sso: This file is the primary report file.

10. ss_summary.sso: Output file that contains all the likelihood components, parameters, derived quantities, total biomass, summary biomass, and catch. This file offers an abridged version of the report file that is useful for quick model evaluation. This file is only available in v.3.30.08.03 and greater.

11. CompReport.sso: Observed and expected composition data in a list-based format.

12. Forecast-report.sso: Output of management quantities and for forecasts.

13. CumReport.sso: This file contains a brief version of the run output, output is appended to current content of file so results of several runs can be collected together. This is useful when a batch of runs is being processed.

14. Covar.sso: This file replaces the standard ADMB ss.cor with an output of the parameter and derived quantity correlations in database format.

15. ss3.par (previously ss.par): This file contains all estimated and fixed parameters from the model run.

16. ss.std, ss.rep, ss.cor etc.: Standard ADMB output files.

17. checkup.sso: Contains details of selectivity parameters and resulting vectors. This is written during the first call of the objective function.

18. Gradient.dat: New for v.3.30, this file shows parameter gradients at the end of the run.

19. rebuild.dat: Output formatted for direct input to Andre Punt's rebuilding analysis package. Cumulative output is output to REBUILD.SS (useful when doing MCMC or profiles).

20. SIS_table.sso: Output formatted for reading into the NMFS Species Information System.

21. Parmtrace.sso: Parameter values at each iteration.

22. posteriors.sso, derived_posteriors.sso, posterior_vectors.sso: Files associated with MCMC.

Starting Stock Synthesis

SS3 is typically run through the command line interface, although it can also be called from another program, R, the Stock Synthesis Interface, or a script file (such as a DOS batch file). SS3 is compiled for Windows, Mac, and Linux operating systems. The memory requirements depend on the complexity of the model you run, but in general, SS3 will run much slower on computers with inadequate memory. See Running Stock Synthesis for additional notes on methods of running SS3.

Communication with the program is through text files. When the program first starts, it reads the file starter.ss, which typically must be located in the same directory from which SS3 is being run. The file starter.ss contains required input information plus references to other required input files, as described in the File Organization section. The names of the control and data files must match the names specified in the starter.ss file. File names, including starter.ss, are case-sensitive on Linux and Mac systems but not on Windows. The echoinput.sso file outputs how the executable reads each input file and can be used for troubleshooting when trying to setup a model correctly. Output from SS3 consists of text files containing specific keywords. Output processing programs, such as Excel, or R can search for these keywords and parse the specific information located below that keyword in the text file.

Converting Files from Stock Synthesis v.3.24

Converting files from version 3.24 to version 3.30 can be performed by using the program ss_trans.exe. This executable takes v.3.24 files as input and will output v.3.30 input and output files. SS_trans executables are available for v.3.30.01 - v.3.30.17. The transitional executable was phased out with v.3.30.18. If a model needs to be converted from v.3.24 to a recent version, one should use the v.3.30.17 ss_trans.exe available from the v.3.30.17 release page on GitHub to convert the files and then any additional adjustments needed between v.3.30.17 and newer versions should be done by hand. To see the changes that need to be made between v.3.30.17 and the latest release of SS3, please see the change log for v.3.30.19 onward as well as the Excel version of the change log for versions prior to v.3.30.19.

The following file structure and steps are recommended for converting model files:

1. Create "transition" folder. Place the 4 main model files (control, data, starter, and forecast) from version SS3 v.3.24 within the transition folder along with the SS3 transition executable (ss_trans.exe). One tip is to use the control.ss_new from the SS3 v.3.24 estimated model rather than the control.ss file which will set all parameter values at the previous estimated maximum likelihood estimated (MLE) parameters. Run the transition executable with phase = 0 within the starter file with the read par file turned off (option 0).

2. Create "converted" folder. Place the ss_new (data.ss_new, control.ss_new, starter.ss_new, forecast.ss_new) files created by the transition executable contained within the "transition" folder into this new folder. Rename the ss_new files to the appropriate suffixes and change the names in the starter.ss file accordingly.

3. Review the control (control.ss_new) file to determine that all model functions converted correctly. The structural changes and assumptions for a couple of the advanced model features are too complicated to convert automatically. See below for some known features that may not convert. When needed, it is recommended to modify the control.ss_new file, the converted control file, for only the features that failed to convert properly.

4. Change the max phase to a value greater than the last phase in which the a parameter is set to estimated within the control file. Run the new v.3.30 executable (ss3.exe) within the "converted" folder using the renamed ss_new files created from the transition executable.

5. Compare likelihood and model estimates between the v.3.24 and v.3.30 model versions.

6. If desired, update to versions of Stock Synthesis > v.3.30.17 by running the new v.3.30 input files with the higher executable.

There are some options that have been substantially changed in v.3.30, which impedes the automatic converting of v.3.24 model files. Known examples of v.3.24 options that cannot be converted, but for which better alternatives are available in v.3.30 are:

1. The use of Q deviations,

2. Complex birth seasons,

3. Environmental effects on spawner-recruitment parameters,

4. Setup of time-varying quantities for models that used the no-longer-available features (e.g., logistic bound constraint).

Starter File

SS3 begins by reading the file starter.ss. The term COND appears in the "Typical Value" column of this documentation (it does not actually appear in the model files), it indicates that the following section is omitted except under certain conditions, or that the factors included in the following section depend upon certain conditions. In most cases, the description in the definition column is the same as the label output to the ss_new files.

Starter File Options (starter.ss)

Value	Options	Description
#C this is a starter comment	Must begin with #C then rest of the line is free form	All lines in this file beginning with #C will be retained and written to the top of several output files
data_ file.dat		File name of the data file
control_ file.ctl		File name of the control file
0	Initial Parameter Values: 0 = use values in control file; and 1 = use ss3.par after reading setup in the control file.	Do not set equal to 1 if there have been any changes to the control file that would alter the number or order of parameters stored in the ss3.par file. Values in ss3.par can be edited, carefully. Do not run ss_trans.exe from a ss3.par from v.3.24.
1	Run display detail: 0 = none other than ADMB outputs; 1 = one brief line of display for each iteration; and 2 = fuller display per iteration.	With option 2, the display shows value of each -logL component for each iteration and it displays where crash penalties are created
1	Detailed age-structure report: 0 = minimal output for data-limited methods; 1 = include all output (with wtatage.ss_new); 2 = brief output, no growth; and 3 = custom output	Option 0 will forgo the writing of the Report file, but the ss_summary file will be written that has minimal derived and estimated quantities. This is a useful option for some data-limited assessment approaches (e.g., XSSS or SSS). Option 1 will write out the full Report file. Option 2 will write out select items in the Report file and will omit some more detailed sections (e.g., numbers-at-age).

Value	Options	Description
COND: Detailed age-structure report = 3 -100 -5 9 11 15 -999		Custom report options: First value: -100 start with minimal items or -101 start with all items; Next Values: A list of items to add or remove where negative number items are removed and positive number items added, -999 to end. The reporting numbers for each item that can be selected or omitted are shown in the Report file next to each section key word.
0	Write 1st iteration details: 0 = omit; and 1 = write detailed intermediate calculations to echoinput.sso during first call.	This output is largely unformatted and undocumented and is mostly used by the developer.
0	Parameter Trace: 0 = omit; 1 = write good iteration and active parameters; 2 = write good iterations and all parameters; 3 = write every iteration and all parameters; and 4 = write every iteration and active parameters.	This controls the output to parmtrace.sso. The contents of this output can be used to determine which values are changing when a model approaches a crash condition. It also can be used to investigate patterns of parameter changes as model convergence slowly moves along a ridge. In order to access parameter gradients option 4 should be selected which will write the gradient of each parameter with respect to each likelihood component

Value	Options	Description
1	Cumulative Report: 0 = omit; 1 = brief; and 2 = full.	Controls reporting to the file Cumreport.sso. This cumulative report is most useful when accumulating summary information from likelihood profiles or when simply accumulating a record of all model runs within the current subdirectory
1	Full Priors: 0 = only calculate priors for active parameters; and 1 = calculate priors for all parameters that have a defined prior.	Turning this option on (1) adds the log likelihood contribution from all prior values for fixed and estimated parameters to the total negative log likelihood. With this option off (0), the total negative log likelihood will include the log likelihood for priors for only estimated parameters.
1	Soft Bounds: 0 = omit; and 1 = use.	This option creates a weak symmetric beta penalty for the selectivity parameters. This becomes important when estimating selectivity functions in which the values of some parameters cause other parameters to have negligible gradients, or when bounds have been set too widely such that a parameter drifts into a region in which it has negligible gradient. The soft bound creates a weak penalty to move parameters away from the bounds.

Value	Options	Description
1	Number of Data Files to Output: 0 = none; As of v.3.30.16, none of the .ss_new files will be produced; 1 = output an annotated replicate of the input data file; 2 = add a second data file containing the model's expected values with no added error. ; and 3+ = add N-2 parametric bootstrap data files.	All output files are sequentially output to data_echo.ss_new and need to be parsed by the user into separate data files. The output of the input data file makes no changes, retaining the order of the original file. Output files 2-N contain only observations that have not been excluded through use of the negative year denotation, and the order of these output observations is as processed by the model. At this time, the tag recapture data is not output to data_echo.ss_new. As of v.3.30.19, the output file names have changed; now a separate file is created for the echoed data (data_echo.ss_new), the expected data values given the model fit (data_expval.ss), and any requested bootstrap data files (data_boot_x.ss where x is the bootstrap number). In versions before v.3.30.19, each of these outputs was printed to a single file called data.ss_new.
8	Turn off estimation: -1 = exit after reading input files; 0 = exit after one call to the calculation routines and production of sso and ss_new files; and <positive value> = exit after completing this phase.	The 0 option is useful for (-1) quickly reading in a messy set of input files and producing the annotated control.ss_new and data_echo.ss_new files, or (0) examining model output based solely on input parameter values. Similarly, the value option allows examination of model output after completing a specified phase. Also see usage note for restarting from a specified phase.
1000	MCMC burn interval	Number of iterations to discard at the start of an MCMC run.
200	MCMC thin interval	Number of iterations to remove between the main period of the MCMC run.

Value	Options	Description
0.0	Jitter: 0 = no jitter done to starting values; and >0 starting values will vary with larger jitter values resulting in larger changes from the parameter values in the control or par file.	The jitter function has been revised with v.3.30. Starting values are now jittered based on a normal distribution with the $\mathrm{pr(P_{MIN})} = 0.1\%$ and the $\mathrm{pr(P_{MAX})} = 99.9\%$. A positive value here will add a small random jitter to the initial parameter values. When using the jitter option, care should be given when defining the low and high bounds for parameter values and particularly -999 or 999 should not be used to define bounds for estimated parameters.
-1	SD Report Start: -1 = begin annual SD report in start year; and <year> = begin SD report this year.	
-1	SD Report End: -1 = end annual SD report in end year; -2 = end annual SD report in last forecast year; and <value> = end SD report in this year.	
2	Extra SD Report Years: 0 = none; and <value> = number of years to read.	In a long time series application, the model variance calculations will be smaller and faster if not all years are included in the SD reporting. For example, the annual SD reporting could start in 1960 and the extra option could select reporting in each decade before then.

COND: If Extra SD report years > 0

1940 1950	Vector of years for additional SD reporting. The number of years need to equal the value specified in the above line (Extra SD Reporting).

Value	Options	Description
0.0001	Final convergence	This is a reasonable default value for the change in log likelihood denoting convergence. For applications with much data and thus a large total log likelihood value, a larger convergence criterion may still provide acceptable convergence
0	Retrospective year: 0 = none; and -x = retrospective year relative to end year.	Adjusts the model end year and disregards data after this year. May not handle time varying parameters completely.
0	Summary biomass min age	Minimum integer age for inclusion in the summary biomass used for reporting and for calculation of total exploitation rate.
1	Depletion basis: 0 = skip; 1 = X*SB0; 2 = X*SB$_{MSY}$; 3 = X*SB$_{styr}$; and 4 = X*SB$_{endyr}$. 5 = X*Dynamic SB0 use tens digit (1-9) to invoke multi-year (up to 9 yrs) use 1 as hundreds digit to invoke log(ratio)	Selects the basis for the denominator when calculating degree of depletion in SB. The calculated values are reported to the SD report. Relative to virgin spawning biomass. Relative to spawning biomass that achieves MSY. Relative to model start year spawning biomass. Relative to spawning biomass in the model end year. Relative to the calculated dynamic SB0.
1	Fraction (X) for depletion denominator	Value for use in the calculation of the ratio for SB$_y$/(X*SB0).

Value	Options	Description
1	SPR report basis: 0 = skip; 1 = use 1-SPR$_{target}$; 2 = use 1-SPR at MSY;	SPR is the equilibrium SB per recruit that would result from the current year's pattern and intensity of F's. The quantities identified by 1, 2, and 3 here are all calculated in the benchmarks section. Then the one specified here is used as the selected.
	3 = use 1-SPR at B$_{target}$; and 4 = no denominator, so report actual 1-SPR values.	Denominator in a ratio with the annual value of (1 - SPR). This ratio (and its variance) is reported to the SD report output for the years selected above in the SD report year selection.
4	Annual F units: 0 = skip; 1 = exploitation rate in biomass; 2 = exploitation rate in numbers; 3 = sum(apical F's by fleet); 4 = population F for range of ages; and 5 = unweighted average F for range of ages.	In addition to SPR, an additional proxy for annual F can be specified here. As with SPR, the selected quantity will be calculated annually and in the benchmarks section. The ratio of the annual value to the selected (see F report basis below) benchmark value is reported to the SD report vector. Options 1 and 2 use total catch for the year and summary abundance at the beginning of the year, so combines seasons and areas. But if most catch occurs in one area and there is little movement between areas, this ratio is not informative about the F in the area where the catch is occurring. Option 3 is a simple sum of the full F's by fleet, so may provide non-intuitive results when there are multi areas or seasons or when the selectivities by fleet do not have good overlap in age. Option 4 is a real annual F calculated as a numbers weighted F for a specified range of ages (read below). The F is calculated as Z-M where Z and M are each calculated an $\ln(N_{t+1}/N_t)$ with and without F active, respectively. The numbers are summed over all biology morphs and all areas for the beginning of the year, so subsumes any seasonal pattern.
COND: If F std reporting ≥ 4		Specify range of ages. Upper age must be less than max age because of incomplete handling of the accumulator age for this calculation.
3 7	Age range if F std reporting = 4.	

Value	Options	Description
1	F report basis: 0 = not relative, report raw values; 1 = use F std value relative to SPR_{target}; 2 = use F std value relative to F_{MSY}; and 3 = use F std value relative to $F_{Btarget}$. use tens digit (1-9) to invoke multi-year (up to 9 yrs) F std use 1 as hundreds digit to invoke log(ratio)	Selects the denominator to use when reporting the F std report values. A new option to allow for the calculation of a multi-year trailing average in F was implemented in v.3.30.16. This option is triggered by appending the number of years to calculate the average across where an input of 1 or 11 would result in the SPR_{target} with no changes. Alternatively a value of 21 would calculate F as SPR_{target} with a 2-year trailing average.
0.01	MCMC output detail: 0 = default; 1 = output likelihood components and associated lambda values; 2 = write report for each mceval; and 3 = make output subdirectory for each MCMC vector.	Specify format of MCMC output. This input requires the specification of two items; the output detail and a bump value to be added to the ln(R0) in the first call to MCMC. A bias adjustment of 1.0 is applied to recruitment deviations in the MCMC phase, which could result in reduced recruitment estimates relative to the MLE when a lower bias adjustment value is applied. A small value, called the "bump", is added to the ln(R0) for the first call to MCMC in order to prevent the stock from hitting the lower bounds when switching from MLE to MCMC. If you wanted to select the default output option and apply a bump value of 0.01 this is specified by 0.01 where the integer value represents the output detail and the decimal is the bump value.
0	Age-length-key (ALK) tolerance level	This effect is disabled in code, enter 0.

Value	Options	Description
COND: Seed Value (i.e., 1234)		Specify a seed for data generation. This feature is not available in versions prior to v.3.30.15 This is an optional input value allowing for the specification of a random number seed value. If you do not want to specify a seed, skip this input line and end the starter file with the check value (3.30).
3.30	Model version check value.	A value of 3.30 indicates that the control and data files are currently in v.3.30 format. A value of 999 indicates that the control and data files are in a previous v.3.24 version. The ss_trans.exe executable should be used and will convert the v.3.24 files the control.ss_new and data_echo.ss_new files to the new format. All ss_new files are in the v.3.30 format, so starter.ss_new has v.3.30 on the last line. The mortality-growth parameter section has a new sequence and v.3.30 cannot read a ss3.par file produced by v.3.24 and earlier, so ensure that read par file option at the top of the starter file is set to 0. The Converting Files from Stock Synthesis v.3.24 section has additional information on model features that may impede file conversion.

End of Starter File

Forecast File

The specification of options for forecasts is contained in the mandatory input file named forecast.ss. See Forecast Module: Benchmark and Forecasting Calculations for additional details.

The term COND appears in the "Typical Value" column of this documentation (it does not actually appear in the model files) and indicates that the following section is omitted except under certain conditions, or that the factors included in the following section depend upon certain conditions. In most cases, the description in the definition column is the same as the label output to the ss_new files.

Forecast File Options (forecast.ss)

Value	Options	Description
1	Benchmarks/Reference Points: 0 = skip/omit; 1 = calculate F_{SPR}, $F_{Btarget}$, and F_{MSY}; 2 = calculate F_{SPR}, F_{MSY}, $F_{0.10}$; and 3 = add F at B_{LIMIT}	SS3 checks for consistency of the forecast specification and the benchmark specification. It will turn on benchmarks if necessary and report a warning.
1	MSY Method: 1 = F_{SPR} as proxy; 2 = calculate F_{MSY}; 3 = $F_{Btarget}$ as proxy or $F_{0.10}$; 4 = $F_{end\ year}$ as proxy; and 5 = F_{MEY}.	Specifies basis for calculating a single population level F_{MSY} value.
COND: MSY Method = 5		
1	Maximum economic yield (MEY) units 1 = dead biomass; 2 = dead biomass without excluded bycatch fleet; 3 = retained biomass; and 4 = profits using price and costs.	

Value	Options	Description
1 0 0 1 -9999 0 0 0	MEY options - Fleet, Cost/F, Price/F, and Include F_{MEY} in Optimization	To calculate the F_{MEY} enter fleet number, the cost per fishing mortality, price per mt, and whether optimization should adjust the fleet's F or keep it at the mean from the benchmark years (0 = no, 1= yes). Take care when scaling the values used for cost/F and price/mt. Units in the example show cost = 0 and price = 1, so it will be identical to MSY in weight. Note, if a fleet's catch is excluded from the F_{MEY} search, its catch or profits are still included in the MSY value using historical F levels from benchmark years.
0.45	SPR_{target}	SS3 searches for the F multiplier that will produce this level of spawning biomass per recruit (reproductive output) relative to the unfished value.
0.40	Relative Biomass Target	SS3 searches for the F multiplier that will produce this level of spawning biomass relative to unfished value. This is not "per recruit" and takes into account the spawner-recruitment relationship.
COND: Benchmarks = 3 -0.25		B_{LIMIT} as a fraction of the B_{MSY} where a negative value will be applied as a fraction of B0
0 0 0 0 0 0 0 0 0 0	Benchmark Years: -999: start year; >0: absolute year; and <= 0: year relative to end year.	Requires 5 pairs of year values over which the mean of derived vectors will be calculated to use in the benchmark (e.g. MSY) calculations. First pair of years is for biology (e.g., growth, natural mortality, maturity, fecundity); second is selectivity; third is relative Fs among fleets; fourth is movement and recruitment distribution; fifth is stock-recruitment (as the parameters, not as derived quantities). If a factor is not time-varying, select the first model year for the beginning year for the factor or else the variance will be artificially reduced.

Value	Options	Description
1	Benchmark Relative F Basis: 1 = use year range; and 2 = set range for relF same as Forecast.	The specification does not affect year range for selectivity and biology.
2	Forecast: -1 = none, no forecast years; 0 = simple, single forecast year calculated; 1 = use F_{SPR}; 2 = use F_{MSY}; 3 = use $F_{Btarget}$ or $F_{0.10}$; 4 = set to mean F scalar for the forecast relative F years below; and 5 = input annual F scalar.	This input is required but is ignored if benchmarks are turned off. This determines how forecast catches are calculated and removed from the population which is separate from the "MSY Method" above. If F_{MSY} is selected, it uses whatever proxy (e.g., F_{SPR} or F_{BTGT}) is selected in the "MSY Method" row.
10	N forecast years (must be >= 1)	At least one forecast year now required if the Forecast option above is >=0 (Note: v.3.24 allowed zero forecast years).
1	F scalar/multiplier	Only used if Forecast option = 5 (input annual F scalar), but is a required line in the forecast file.

There are 2 options for entering Forecast Years:

Option 1:	This approach for forecast year ranges is no longer recommended because blocks, random effects, and other time-varying parameter changes can now operate on forecast years and the new approach provides better control averaging.

Value	Options	Description
0 0 0 0 0 0	Enter 6 Forecast Year Values	To continue to use this pre-v.3.20.22 approach, enter 6 values: beginning and ending years for selectivity, relative Fs, and recruitment distribution. These are used to create means over the specified range of years. Values can be entered as the actual year, -999 for start year, or values of 0 or -integer to be relative endyr. It is important to note: – Relative F for bycatch only fleets is scaled just like other fleets. – For selectivity averaging with the new approach the method code is "1", whereas with the old Forecast Selectivity Option, the code was "1" for using time-varying parameters. SS3 accounts for this change internally. – Whenever calculating means, the calculated mean will have artificially low variance than if a minimal range of years is selected.
0	Forecast Selectivity Option: 0 = forecast selectivity means from year range; and 1 = forecast selectivity from annual time-varying parameters.	Determines selectivity used in the forecast years. Selecting 1 will allow for application of time-varying selectivity parameters (e.g., random walk) to continue into the forecast period. This setting is not included in Option 2.
Option 2:	To use the new approach, enter -12345 and omit entry of the Forecast Selectivity Option.	
-12345	Invoke New Forecast Format	Biology and selectivity vectors are updated annually in the forecast according to their time-varying parameters. Be sure to check the end year of the blocks and the deviation vectors. Input in this section directs the creation of means over historical years to override any time-varying changes. To invoke taking the mean of a range of historical recruitments after all adjustments and deviations were applied, see the Base recruitment in forecast option. See the Example New Forecast Format Input below.

Value	Options		Description	
Example New Forecast Format Input:				
Factor	Method	Start Year	End Year	
1	1	2002	2003	# natural mortality
4	1	2016	2018	# recruitment distribution
10	1	-999	0	# selectivity
11	1	-3	0	# relative F
12	1	2006	2014	# recruitment
-9999	-1	-1	-1	
	Factor		Factors implemented thus far. Terminate with -9999.	
	1 = natural mortality (M);			
	4 = recruitment distribution;			
	5 = migration;			
	10 = selectivity;			
	11 = relative F			
	12 = recruitment			
	Method			
	0 (or omitted) = continue using time_vary parameters;			
	1 = use means of derived factor;			
	2 (future) = means parameter then apply as if time_vary			
	Start Year		Enter the actual year or values of 0, -999 to be styr, or -integer to be relative endyr.	
	End Year		Enter the actual year or values of 0 or -integer to be relative endyr.	

Value	Options	Description
1	Control Rule Method: 0 = none (additional control rule inputs will be ignored); 1 = catch as function of SSB, buffer on F; 2 = F as function of SSB, buffer on F; 3 = catch as function of SSB, buffer on catch (U.S. West Coast groundfish approach); and 4 = F is a function of SSB, buffer on catch.	Used to apply reductions ("buffer") to either the catch or F based on the control rule during the forecast period. The buffer value is specified below via the Control Rule Buffer.
0.40	Control Rule Inflection	Relative biomass level to unfished biomass above which F is constant at control rule F. If set to -1 the ratio of B_{MSY} to the unfished spawning biomass will automatically be used.$_{target}$.
0.10	Control Rule Cutoff	Relative biomass level to unfished biomass below which F is set to 0 (management threshold).
0.75	Control Rule Buffer (multiplier between 0-1.0 or -1)	Control rule catch or F_{target} as a fraction of selected catch or F_{MSY} proxy. The buffer will be applied to reduce catch from the estimated overfishing limit. The buffer value is a value between 0-1.0 where a value of 1.0 would set catch equal to the overfishing limit. As example if the buffer is applied to catch (Control Rule option 3 or 4 above) the catch will equal the buffer times the overfishing limit. Alternatively a value of -1 will allow the user to input a forecast year specific control rule fraction (added in v.3.30.13).

Value	Options	Description
COND -1: Conditional input for annual control rule buffer 2019 0.8 2020 0.6 2021 0.5 -9999 0		Year and control rule buffer value. Can enter a value for each year, or starting sequence of years. The final control rule buffer value will apply to all sequent forecast years.
3	Number of forecast loops 1 = OFL only; 2 = ABC control rule and buffers; 3 = set catches equal to control rule or input catch and redo forecast implementation error.	SS3 sequentially goes through the forecast up to three times.
3	First forecast loop with stochastic recruitment	If this is set to 1 or 2, then OFL and ABC will be calculated as if there was perfect knowledge about future recruitment deviations. If running a long forecast (e.g., 10-100 years) its recommended to run without recruitment deviations because running long forecasts where recruitment deviations aren't turned on until loop 3 may have poor results (e.g., crashed stock), especially if below mean forecast recruitment is assumed (via Base recruitment in forecast option, next input line).

Value	Options	Description
1	Base recruitment in forecast: 0 = spawner recruit curve; 1 = value*(spawner recruit curve); 2 = value*(virgin recruitment); 3 = deprecated; and 4 = mean recruitment from Forecast Year range above, recruitment distribution not affected.	This option controls the base recruitment (to which deviations are applied) in the forecast, or taking the mean of a range of historical recruitments after all adjustments and deviations were applied. For options 1 and 2, the next value read is a scalar applied to the base. Option 4 requires the user set the forecast recruitment deviation phase to negative (specifically -1 to get constant mean in MCMC) and the last year of recruitment deviations is the end year.
0.7	Scalar/multiplier applied to base	Scalar is ignored unless option 1 and 2 is selected.
0	Not used	
2015	First year for caps and allocations	Should be after years with fixed inputs.
0	Implementation Error	The standard deviation of the log of the ratio between the realized catch and the target catch in the forecast. (set value > 0.0 to cause implementation error deviations to be an estimated parameter that will add variance to forecast).
0	Do West Coast Groundfish Rebuilder Output: 0 = omit U.S. West Coast rebuilder output; and 1 = do abbreviated U.S. West Coast rebuilder output	Creates a rebuild.dat file to be used for U.S. West Coast groundfish rebuilder program.

Value	Options	Description
2004	Rebuilder catch (Year Declared): > 0 = year first catch should be set to zero; and -1 = set to 1999.	Input line is required even if Rebuilder = 0, specified in the line above.
2004	Rebuilder start year (Year Initial): > 0 = year for current age structure; and -1 = set to end year +1.	Input line is required even if Rebuilder = 0, specified two line above.
1	Fleet Relative F: 1 = use first-last allocation year; and 2 = read season(row) x fleet (column) set below.	
2	Basis for maximum forecast catch: 2 = total catch biomass; 3 = retained catch biomass; 5 = total catch numbers; and 6 = retained total numbers.	The maximum basis for forecasted catch will be implemented for the for the "First year for caps and allocations" selected above. The maximum catch (biomass or numbers) by fleet is specified below on the "Maximum total forecast catch by fleet" line.

COND 2: Conditional input for fleet relative F (Enter: Season, Fleet, Relative F)

1 1 0.6	Fleet allocation by relative F fraction.	The fraction of the forecast F value. For a multiple area model user must define a fraction for each fleet and each area. The total fractions must sum to one over all fleets and areas.
1 2 0.4		
-9999 0 0	Terminator line	

Value	Options	Description
1 50 -9999 -1	Maximum total forecast catch by fleet (in units specified above total catch/numbers, retained catch/numbers)	Enter fleet number and its maximum value. Last line of the entry must have fleet number = -9999.
-9999 -1	Maximum total catch by area -1 = no maximum	Enter area number and its max. Last line of the entry must have area number = -9999.
1 1 -9999 -1 COND: if N allocation groups is > 0 2002 1 -9999 1	Fleet assignment to allocation group Allocation to each group for each year of the forecast	Enter list of fleet number and its allocation group number if it is in a group. Last line of the entry must have fleet number = -9999. Enter a year and the allocation fraction to each group for that year. SS3 will fill those values to the end of the forecast, then read another year from this list. Terminate with -9999 in year field. Annual values are rescaled to sum to 1.0.
-1	Basis for forecast catch: -1 = Read basis with each observation, allows for a mixture of dead, retained, or F basis by different fleets for the fixed catches below; 2 = Dead catch (retained + discarded); 3 = Retained catch; and 99 = Input apical F (the apical F value for the model years can be found in the EXPLOITATION section in the Report file).	The dead or retained value in the forecast catch inputs will be interpreted in terms of numbers or biomass based on the units of the input catch for each fleet.

Value	Options	Description
COND: == -1	Forecasted catches - enter one line per number of fixed forecast year catch	
2012 1 1 1200 2	Year & Season & Fleet & Catch or F value & Basis	
2013 1 1 1400 3	Year & Season & Fleet & Catch or F value & Basis	
-9999 0 0 0 0	Indicates end of inputted catches to read	
COND: > 0	Forecasted catches - enter one line per number of fixed forecast year catch	
2012 1 1 1200	Year & Season & Fleet & Catch or F value	
2013 1 1 1200	Year & Season & Fleet & Catch or F value	
-9999 0 0 0	Indicates end of inputted catches to read	
999	End of Input	
	End of Forecast File	

Including a New Fleet in the Forecast

As of v.3.30.16 users can have a forecast fleet without catches during the modeled period. Previously, fleets in the forecast period were required to have input catches at some amount during the modeled period. SS3 now has capability to have a fleet with no input catches during the modeled period that could be used as a fleet during the forecast.

Benchmark Calculations

This feature of SS3 is designed to calculate an equilibrium fishing rate intended to serve as a proxy for the fishing rate that would provide maximum sustainable yield (F_{MSY}). Then in the forecast module these fishing rates can be used in the projections.

Four reference points can be calculated by SS3. The first is the estimate of F_{MSY} within the model, while the other reference points use proxies or an alternative estimated point.

- F_{MSY}: Search for the F that produces maximum equilibrium (e.g., dead catch).

- F_{SPR}: Search for the F that produces spawning biomass per recruit this is a specific fraction, termed SPR_{target}, of spawning biomass per recruit under unfished conditions. Note that this is in relative terms so it does not take into account the spawner-recruit relationship.

- $F_{Btarget}$: Search for the F that produces an absolute spawning biomass that is a specified fraction, termed relative biomass target, of the unfished spawning biomass. Note that this is in absolute terms so takes into account the spawner-recruit relationship.

- $F_{0.10}$: Search for the F that produces a slope in yield per recruit, dY/dF, that is 10% of the slope at the origin. Note that with SS3, this option is mutually exclusive with $F_{Btarget}$. Only one will be calculated and the one that is calculated can serve as the proxy for F_{MSY} and forecasting. The $F_{0.1}$ search can fail if bycatch fleets are used and the bycatch setting includes the fleet's catch in the catch to be optimized.

Estimation
Each of the potential reference points is calculated by searching across a range of F multiplier levels, calculating equilibrium biomass and catch at that F, using Newton-Raphson method to calculate a better F multiplier value, and iterating a fixed number of times to achieve convergence on the desired level.

Calculations
The calculation of equilibrium biomass and catch uses the same code that is used to calculate the virgin conditions and the initial equilibrium conditions. This equilibrium

calculation code takes into account all morph, timing, biology, selectivity, and movement conditions as they apply while doing the time series calculations. You can verify this by running SS3 to calculate F_{MSY} then hardwire initial F to equal this value, use the F_method approach 2 so each annual F is equal to F_{MSY} and then set forecast F to be the same F_{MSY}. Then run SS3 without estimation and no recruitment deviations. You should see that the population has an initial equilibrium abundance equal to B_{MSY} and stays at this level during the time series and forecast.

Catch Units

For each fleet, SS3 always calculates catch in terms of biomass (mt) and numbers (1000s) for encountered (selected) catch, dead catch, and retained catch. These three categories differ only when some fleets have discarding or are designated as a bycatch fleet. SS3 uses total dead catch biomass as the quantity that is principally reported and the quantity that is optimized when searching for F_{MSY}. The quantity "dead catch" may occasionally be referred to as "yield".

Biomass Units

The principle measure of fish abundance, for the purpose of reference point calculation, is female reproductive output. This is referred to as SSB (spawning stock biomass) and sometimes just "B" because the typical user settings have one unit of reproductive output (fecundity) per kg of mature female biomass. So when the output label says B_{MSY}, this is actually the female reproductive output at the proxy for F_{MSY}.

Fleet Allocation

An important concept for the reference point calculation is the allocation of fishing rate among fleets. Internally, this is benchmark years relative F (Bmark_relF(f, s)) and it is the fraction of the F multiplier assigned to each fleet, f and season, s. The value, F_multiplier * Bmark_relF(f, s), is the F level for a particular fleet in a particular season and for the age that has a selectivity of 1.0. Other ages will have different F values according to their selectivity.

- The Bmark_relF values can be calculated by SS3 from a range of years specified in the input for Benchmark Years or it can be set to be the same as the Forecast_RelF, which in turn can be based on a range of years or can be input as a set of fixed values.

- The biology years selected for the Bmark_relF calculations can have an effect on the standard deviation estimated, even in models with no time-varying biology. It is recommended to use the first model year except when biology is time-varying.Note that when biology is time-varying, the fecundity vector, which is used to calculate SSB, will be updated in every iteration for every year that has time-varying biology.

- Note that for Bycatch Fleets, the F's calculated by application of Bmark_relF for a bycatch fleet can be overridden by a F value calculated from a range of years or a

fixed F value that is input by the user. If such an override is selected for a bycatch fleet, that F value is not adjusted by changes to the F multiplier. This allows the user to treat a bycatch fleet as a constant background F while the optimal F for other fleets is sought. Also for bycatch fleets, there is user control for whether or not the dead catch from the bycatch fleet is included in the total dead catch that is optimized when searching for F_{MSY}.

Virgin vs. Unfished

The concept of unfished spawning biomass, SSB_unf, is important to the reference points calculations. Unfished spawning biomass can be potentially different than virgin spawning biomass, SSB_virgin.

- Virgin spawning biomass is calculated from the parameter values associated with the start year of the model configuration and it serves as the basis from which the population model starts and the basis for calculation of stock depletion.

- Unfished spawning biomass can be calculated for any year or range of years, so can change over time as R0, steepness, or biological parameters change.

- In the reference points calculation, the Benchmark Years input specifies the range of time over which the mean of various quantities are taken to calculate the reference points. For biology, selectivity, F's, and movement the mean values are the year-specific derived quantities. But for the stock-recruitment parameters (R0 and steepness), the mean of the parameter values themselves is calculated over time.

- During the time series or forecast, the current year's unfished spawning output (SSB_unf) is used as the basis for the spawner-recruitment curve against which deviations from the spawner-recruitment curve are applied. So, if R0 is made time-varying, then the spawner-recruit curve itself is changed. However, if the regime shift parameter is time-varying, then this is an offset from the spawner-recruitment curve and not a change in the curve itself. Changes in R0 will change year-specific reference points and change the expected value for annual recruitments, but changes in regime shift parameter only change the expected value for annual recruitments.

- In reporting the time series of depletion level, the denominator can be based on virgin spawning output (SSB_virgin) or B_{MSY}. Note that B_{MSY} is based on unfished spawning output (SSB_unf) for the specified range of Benchmark years, not on SSB_virgin.

Forecast Recruitment Adjustment

Recruitment during the forecast years sometimes needs to be set at a level other than that determined by the spawner-recruitment curve. One way to do this is by an environmental

or block effect on the regime shift parameter. A more straightforward approach is now provided by the special forecast recruitment feature described here. There are 4 options provided for this feature. These are:

- 0 = Do nothing: this is the default and will invoke no special treatment for the forecast recruitments.

- 1 = Multiplier on spawner-recruitment: the expected recruitment from the stock recruitment relationship is multiplied by this factor.

  – This is a multiplier, so null effect comes from a value of 1.0.

  – The order of operations is to apply the SRR, then the regime effect, then this special forecast effect, then bias adjustment, then the deviations.

  – In the spawner recruit output of the report.sso there are 4 recruitment values stored.

- 2 = Multiplier on virgin recruitment: The virgin recruitment is multiplied by this factor.

  – This is a multiplier, so null effect comes from a value of 1.0.

  – The order of operations is to apply any environmental or block effects to R0, then apply the special forecast effect, then bias adjustment, then the deviations.

  – Note that environmental or block effects on R0 are rare and are different than environment or block effects on the regime parameter.

- 3 = Mean recent recruitment: calculate the mean recruitment and use it during the forecast period.

  – Note that bias adjustment is not applied to this mean because the values going into the mean have already been bias adjusted.

This feature affects the expected recruitment in all years after the last year of the main recruitment deviations. This means that if the last year of main recruitment deviations is before end year, then the last few recruitments, termed "late", are also affected by this forecast option. For example, option 3 would allow you to set the last 2 years of the time series and all forecast years to have recruitment equal to the mean recruitment for the last 10 years of the main recruitment era.

Data File

Overview of Data File

1. Dimensions (years, ages, number of fleets, number of surveys, etc.)

2. Fleet and survey names, timing, etc.

3. Catch data (biomass or numbers)

4. Discard totals or rate data

5. Mean body weight or mean body length data

6. Length composition set-up

7. Length composition data

8. Age composition set-up

9. Age imprecision definitions

10. Age composition data

11. Mean length-at-age or mean bodyweight-at-age data

12. Generalized size composition (e.g., weight frequency) data

13. Environmental data

14. Tag-recapture data

15. Stock composition (e.g., morphs identified by otolith microchemistry) data

16. Selectivity observations (new placeholder, not yet implemented)

Units of Measure

The normal units of measure are as follows:

- Catch biomass - metric tons

- Body weight - kilograms

- Body length - usually in centimeters, weight-at-length parameters must correspond to the units of body length and body weight

- Survey abundance - any units if catchability (Q) is freely scaled; metric tons or thousands of fish if Q has a quantitative interpretation

- Output biomass - metric tons

- Numbers - thousands of fish, because catch is in metric tons and body weight is in kilograms

- Spawning biomass - metric tons of mature females if eggs/kg = 1 for all weights; otherwise has units that are based on the user-specified fecundity

Time Units

- Spawning:

  - Happens once per year at a specified date (in real months, 1.0 - 12.99). To create multiple spawning events per year, change the definition of a year (e.g., call a 6 month period a "year" when there are two spawning events about 6 months apart). However, revising the definition of year affects assignment of fish age, so it should not be used if age data are included.

- Recruitment:

  - Occurs at specified recruitment events that occur at user-specified dates (in real months, 1.0 - 12.99).

  - There can be one to many recruitment events across a year; each producing a platoon as a portion of the total recruitment.

  - A settlement platoon enters the model at age 0 if settlement is between the time of spawning and the end of the year; it enters at age 1 if settlement is after the first of the year; these ages at settlement can be overridden in the settlement setup.

- Timing

  - All fish advance to the next older integer age on January 1, no matter when they were born during the year. Consult with your ageing lab to assure consistent interpretation.

- Parameters:

- Time-varying parameters are allowed to change annually, not seasonally.

- Rates like growth and mortality are per year.

Seasons

Seasonal quantities in the model are calculated and treated in the following methods:

- Seasons are the time step during which constant rates apply.

- Catch and discard amounts are per season and F is calculated per season.

- The year can have just one annual season, or be subdivided into seasons of unequal length.

- Season duration is input in real months (1.0 - 12.99) and is converted into fractions of an annum. Annual rate values are multiplied by the per annum season duration.

- If the sum of the input season durations do not equal 12.0, then the input durations are divided by the total duration input to rescale season duration to equal 1.0.

- Allows for a special situation in which the year could be only 0.25 in duration (e.g., seasons as years) so that spawning and time-varying parameters can occur more frequently. Other durations are also possible (e.g., where a model year is really only 6 months). Note that real month inputs should always have the range 1.0 - 12.99, even in cases where a model year is not a true year. See the continuous seasonal recruitment section for more information on how to set up models where the model year duration is different than a true year.

Subseasons and Timing of Events

The treatment of subseasons in SS3 provide more precision in the timing of events compared to earlier model versions. In early versions, v.3.24 and before, there was effectively only two subseasons per season because the age-length-key (ALK) for each observation used the mid-season mean length-at-age and spawning occurred at the beginning of a specified season.

Time steps can be broken into subseason and the ALK can be calculated multiple times over the course of a year:

ALK	ALK*	ALK*	ALK	ALK*	ALK
Subseason 1	Subseason 2	Subseason 3	Subseason 4	Subseason 5	Subseason 6

ALK* only re-calculated when there is a survey that subseason

- Even number (min = 2) of subseasons per season (regardless of season duration):

    - Two subseasons will mimic v.3.24

    - Specifying more subseasons will give finer temporal resolution, but will slow the model down, the effect of which is mitigated by only calculating growth as needed.

- Survey timing is now cruise-specific and specified in units of months (e.g., April 15 = 4.5; possible inputs are 1.0 to 12.99).

    - ss_trans.exe will convert year, season in v.3.24 format to year, real month in v.3.30 format.

- Survey integer season and spawn integer season assigned at run time based on real month and season duration(s).

- Growth and the age-length-key (ALK) is calculated at the beginning and mid-season or when there is a defined subseason with data observation.

- Fishery body weight uses mid-subseason growth.

- Survey body weight and size composition is calculated using the nearest subseason.

- Reproductive output now has specified spawn timing (in months fraction) and interpolates growth to that timing.

- Survey numbers calculated at cruise survey timing using e^{-z}.

- Continuous Z for entire season. Same as applied in version v.3.24.

Terminology

The term COND appears in the "Typical Value" column of this documentation (it does not actually appear in the model files), it indicates that the following section is omitted except under certain conditions, or that the factors included in the following section depend upon

certain conditions. In most cases, the description in the definition column is the same as the label output to the ss_new files.

Model Dimensions

Value	Description
#V3.30.XX.XX	Model version number. This is written by SS3 in the new files and a good idea to keep updated in the input files.
#C data using new survey	Data file comment. Must start with #C to be retained then written to top of various output files. These comments can occur anywhere in the data file, but must have #C in columns 1-2.
1971	Start year
2001	End year
1	Number of seasons per year
12	Vector with the number of months in each season. These do not need to be integers. Note: If the sum of this vector is close to 12.0, then it is rescaled to sum to 1.0 so that season duration is a fraction of a year. If the sum is not equal to 12.0, then the entered values are summed and rescaled to 1. So, with one season per year and 3 months per season, the calculated season duration will be 0.25, which allows a quarterly model to be run as if quarters are years. All rates in SS3 are calculated by season (growth, mortality, etc.) using annual rates and season duration.
2	The number of subseasons. Entry must be even and the minimum value is 2. This is for the purpose of finer temporal granularity in calculating growth and the associated age-length key.
1.5	Spawning month; spawning biomass is calculated at this time of year (1.5 means January 15) and used as basis for the total recruitment of all settlement events resulting from this spawning.
2	Number of sexes: 1 = current one sex, ignore fraction female input in the control file; 2 = current two sex, use fraction female in the control file; and

-1	= one sex and multiply the spawning biomass by the fraction female in the control file.
20	Number of ages. The value here will be the plus-group age. SS3 starts at age 0.
1	Number of areas
2	Total number of fishing and survey fleets (which now can be in any order).

Fleet Definitions

The catch data input has been modified to improve the user flexibility to add/subtract fishing and survey fleets to a model set-up. The fleet setup input is transposed so each fleet is now a row. Previous versions (v.3.24 and earlier) required that fishing fleets be listed first followed by survey only fleets. In SS3 all fleets have the same status within the model structure and each has a specified fleet type (except for models that use tag recapture data, this will be corrected in future versions). Available types are; catch fleet, bycatch only fleet, or survey.

Inputs that define the fishing and survey fleets:

2	Number of fleets which includes survey in any order				
Fleet Type	Timing	Area	Catch Units	Catch Mult.	Fleet Name
1	-1	1	1	0	FISHERY1
3	1	1	2	0	SURVEY1

Fleet Type
Define the fleet type (e.g., fishery fleet, survey fleet):

- 1 = fleet with input catches;

- 2 = bycatch fleet (all catch discarded) and invoke extra input for treatment in equilibrium and forecast;

- 3 = survey: assumes no catch removals even if associated catches are specified below. If you would like to remove survey catch set fleet type to option = 1 with specific month timing for removals (defined below in the "Timing" section); and

- 4 = predator (M2) fleet that adds additional mortality without a fleet F (added in v.3.30.18). Ideal for modeling large mortality events such as fish kills or red tide. Requires additional long parameter lines for a second mortality component (M2) in the control file after the natural mortality/growth parameter lines (entered immediately after the fraction female parameter line).

Timing
Timing for data observations:

- Fishery options:

    - -1: catch is treated as if it occurred over the whole season. Stock Synthesis may change the CPUE data to occur in the middle of the season if it is specified otherwise (i.e., the CPUE observations may have a different month in the data_echo.ss_new file). A user can override this assumption for specific data observations (e.g., length or age) by specifying a month. This option works well for fisheries where fishing is spread throughout the year.

    - 1: The fleet timing is not used and only the month value associated with each observation is relevant. This option works well for pulse fisheries that occurs over a small subset of months.

- Survey option, 1: The fleet timing is not used and only the month value associated with each observation is relevant (e.g., month specification in the indices of abundance or the month for composition data). This input should always be used for surveys.

Area
An integer value indicating the area in which a fleet operates.

Catch Units
Ignored for survey fleets, their units are read later:

- 1 = biomass (in metric tons); and

- 2 = numbers (thousands of fish).

See Units of Measure for more information.

Catch Multiplier
Invokes use of a catch multiplier, which is then entered as a parameter in the mortality-growth parameter section. The estimated value or fixed value of the catch multiplier is used to adjust the observed catch: Invokes use of a catch multiplier, which is

then entered as a parameter in the mortality-growth parameter section. The estimated value or fixed value of the catch multiplier is used to adjust the observed catch:

- 0 = No catch multiplier used; and

- 1 = Apply a catch multiplier which is defined as an estimable parameter in the control file after the cohort growth deviation in the biology parameter section. The model's estimated retained catch will be multiplied by this factor before being compared to the observed retained catch.

A catch multiplier can be useful when trying to explore historical unrecorded catches or ongoing illegal and unregulated catches. The catch multiplier is a full parameter line in the control file and has the ability to be time-varying.

Bycatch Fleets

The option to include bycatch fleets was introduced in v.3.30.10. This is an optional input and if no bycatch is to be included in to the catches this section can be ignored.

A fishing fleet is designated as a bycatch fleet by indicating that its fleet type is 2. A bycatch fleet creates a fishing mortality, same as a fleet of type 1, but a bycatch fleet has all catch discarded so the input value for retained catch is ignored. However, an input value for retained catch is still needed to indicate that the bycatch fleet was active in that year and season. A catch multiplier cannot be used with bycatch fleets because catch multiplier works on retained catch. SS3 will expect that the retention function for this fleet will be set in the selectivity section to type 3, indicating that all selected catch is discarded dead. It is necessary to specify a selectivity pattern for the bycatch fleet and, due to generally lack of data, to externally derive values for the parameters of this selectivity.

All catch from a bycatch fleet is discarded, so one option to use a discard fleet is to enter annual values for the amount (not proportion) that is discarded in each time step. However, it is uncommon to have such data for all years. An alternative approach that has been used principally in the U.S. Gulf of Mexico is to input a time series of effort data for this fleet in the survey section (e.g., effort is a "survey" of F, for example, the shrimp trawl fleet in the Gulf of Mexico catches and discards small finfish and an effort time series is available for this fleet) and to input in the discard data section an observation for the average discard over time using the super year approach. Another use of bycatch fleet is to use it to estimate effect of an external source of mortality, such as a red tide event. In this usage there may be no data on the magnitude of the discards and SS3 will then rely solely on the contrast in other data to attempt to estimate the magnitude of the red tide kill that occurred. The benefit of doing this as a bycatch fleet, and not a block on natural mortality, is that the selectivity of the effect can be specified.

Bycatch fleets are not expected to be under the same type of fishery management controls as the retained catch fleets included in the model. This means that when SS3 enters into the reference point equilibrium calculations, it would be incorrect to have SS3 re-scale the magnitude of the F for the bycatch fleet as it searches for the F that produces, for example, F35%. Related issues apply to the forecast. Consequently, a separate set of controls is provided for bycatch fleets (defined below). Input is required for each fleet designated as fleet type = 2.

If a fleet above was set as a bycatch fleet (fleet type = 2), the following line is required:

Bycatch fleet input controls:

a: Fleet Index	b: Include in MSY	c: Fmult	d: F or First Year	e: Last Year	f: Not used
2	2	3	1982	2010	0

The above example set-up defines one fleet (fleet number 2) as a bycatch fleet with the dead catch from this fleet to not be included in the search for MSY (b: Include in MSY = 2). The level of F from the bycatch fleet in reference point and forecast is set to the mean (c: Fmult = 3) of the estimated F for the range of years from 1982-2010.

Fleet Index

Fleet number for which to include bycatch catch. Fleet number is assigned within the model based on the order of listed fleets in the Fleet Definition section. If there are multiple bycatch fleets, then a line for each fleet is required in the bycatch section.

Include in MSY

The options are:

- 1 = deadfish in MSY, ABC, and other benchmark and forecast output; and

- 2 = omit from MSY and ABC (but still include the mortality).

Fmult

The options are:

- 1 = F multiplier scales with other fleets;

- 2 = bycatch F constant at input value in column d; and

- 3 = bycatch F from range of years input in columns d and e.

F or First Year

The specified F or first year for the bycatch fleet.

F or Last Year

The specified F or last year for the bycatch fleet.

Not Used

This column is not yet used and is reserved for future features.

Bycatch Fleet Usage Instructions and Warnings

When implementing a bycatch fleet, changes to both the data and control file are needed.

The needed changes to the data file are:

1. Fleet type - set to value of 2.

2. Set bycatch fleet controls per information above.

3. Catch input - you must enter a positive value for catch in each year/season that you want a bycatch calculated. The entered value of catch will be ignored by SS3, it is just a placeholder to invoke creating an F.

   (a) Initial equilibrium - you may want to enter the bycatch amount as retained catch for the initial equilibrium year because there is no option to enter initial equilibrium discard in the discard section.

4. Discard input - It is useful, but not absolutely necessary, to enter the amount of discard to assist SS3 in estimating the F for the bycatch fleet.

5. Survey input - It is useful, but not absolutely necessary, to enter the effort time series by the bycatch fleet to assist SS3 in estimating the annual changes in F for the bycatch fleet.

The needed changes to the control file are:

1. The F method must be set to 2 in order for SS3 to estimate F with having information on retained catch.

2. Selectivity -

   (a) A selectivity pattern must be specified and fixed (or estimated if composition data is provided).

(b) The discard column of selectivity input must be set to a value of 3 to cause all catch to be discarded.

In v.3.30.14 it was identified that there can be an interaction between the use of bycatch fleets and the search for the $F_{0.1}$ reference point which may results in the search failing. Changes to the search feature were implemented to make the search more robust, however, issue may still be encountered. In these instances it is recommended to not select the $F_{0.1}$ reference point calculation in the forecast file.

Predator Fleets

Introduced in v.3.30.18, a predator fleet provides the capability to define an entity as a predator that adds additional mortality (M2, i.e., the predation mortality) to the base natural mortality. This new capability means that previous use of bycatch fleets to mimic predators (or fish kills, e.g., due to red tide) will no longer be necessary. The problem with using a bycatch fleet as a predator was that it still created an "F" that was included in the reporting of total F even if the bycatch was not included in the MSY search.

For each fleet that is designated as a predator, a new parameter line is created in the mortality-growth (MGparm) section in the control file. This parameter will have the label M2_pred1, where the "1" is the index for the predator (not the index of the fleet being used as a predator). More than one predator can be included. If the model has > 1 season, it is normal to expect M2 to vary seasonally. Therefore, only if the number of seasons is greater than 1, follow each M2 parameter with number of season parameters to provide the seasonal multipliers. These are simple multipliers times M2, so at least one of these needs to have a non-estimated value. The set of multipliers can be used to set M2 to only operate in one season if desired. If there is more than one predator fleet, each will have its own seasonal multipliers. If there is only 1 season in the model, then no multiplier lines are included.

Three types of data relevant to M2 can be input:

- Total kill (as discard in the data file): M2 is a component of Z, so M2/Z can be used to calculate the amount of the total kill that is attributable to M2. This is completely analogous to calculating catch for the fishing fleets. The total kill (e.g., consumption) is output to the discard array. If data on the total kill by the M2 predator is available, it can be input as observed "discard" for this fleet and thus included in the total logL to estimate the magnitude of the M2 parameter.

- Predator effort (as a survey index in the data file): M2 is a rate analogous to F, so the survey of F approach (survey units = 2) can be used to input predator abundance as an indicator of the "effort" that produced the M2. Like all surveys, this

survey of M2 will also need a Q specification. Note that in the future we can explore improved options for this Q.

- Predated age-length composition (as length or age composition data in the data file): M2 "eats" the modeled fish, so gut contents or other sources may have size and/or age composition data which may be input to estimate selectivity of the M2 source.

With the input of data on the time series of total kill or predator effort, it should be possible to estimate annual deviations around the base M2 for years with data. If the M2 time series is instead driven by environmental data, then also including data on kill or effort can provide a means to view consistency between the environmental time series and the additional data sets. Output of M2 is found in a report.sso section labeled predator (M2). In the example below, the M2 seasonal multiplier was defined to have random deviations by year. This allowed multipliers plus M2 itself to closely match the input consumption amounts (288 mt of consumption per season, the fit can be examined by looking at the discard output report).

Catch

After reading the fleet-specific indicators, a list of catch values by fleet and season are read in by the model. The format for the catches is year and season that the catch is attributed to, fleet, a catch value, and a year-specific catch standard error. Only positive catches need to be entered, so there is no need for records corresponding to all years and fleets. To include an equilibrium catch value for a fleet and season, the year should be noted as -999. For each non-zero equilibrium catch value included, a short parameter line is required in the initial F section of the control file.

There is no longer a need to specify the number of records to be read; instead the list is terminated by entering a record with the value of -9999 in the year field. The updated list based approach extends throughout the data file (e.g., catch, length- and age-composition data), the control file (e.g., lambdas), and the forecast file (e.g., total catch by fleet, total catch by area, allocation groups, forecasted catch).

In addition, it is possible to collapse the number of seasons. So, if a season value is greater than the number of seasons for a particular model, that catch is added to the catch for the final season. This is one way to easily collapse a seasonal model into an annual model. The alternative option is to the use of season = 0. This will cause SS3 to distribute the input value of catch equally among the number of seasons. SS3 assumes that catch occurs continuously over seasons and hence is not specified as month in the catch data section. However, all other data types will need to be specified by month.

The format for a 2 season model with 2 fisheries looks like the table below. Example is sorted by fleet, but the sort order does not matter. In data.ss_new, the sort order is fleet, year, season.

Catches by year, season for every fleet:

Year	Season	Fleet	Catch	Catch SE
-999	1	1	56	0.05
-999	2	1	62	0.05
1975	1	1	876	0.05
1975	2	1	343	0.05
...	...	...	...	...
...	...	...	...	...
-999	1	2	55	0.05
-999	2	2	22	0.05
1975	1	2	555	0.05
1975	2	2	873	0.05
...	...	...	...	...
...	...	...	...	...
-9999	0	0	0	0

- Catch can be in terms of biomass or numbers for each fleet, but cannot be mixed within a fleet.

- Catch is retained catch (aka landings). If there is discard also, then it is handled in the discard section below. This is the recommended setup which results in a model estimated retention curve based upon the discard data (specifically discard composition data). However, there may be instances where the data do not support estimation of retention curves. In these instances catches can be specified as all dead (retained + discard estimates).

- If there are challenges to estimating discards within the model, catches can be input as total dead without the use of discard data and retention curves.

- If there is reason to believe that the retained catch values underestimate the true catch, then it is possible in the retention parameter set up to create the ability for the model to estimate the degree of unrecorded catch. However, this is better handled with the new catch multiplier option.

Indices

Indices are data that are compared to aggregate quantities in the model. Typically the index is a measure of selected fish abundance, but this data section also allows for the index to be related to a fishing fleet's F, or to another quantity estimated by the model. The first section of the "Indices" setup contains the fleet number, units, error distribution, and whether additional output (SD Report) will be written to the Report file for each fleet that has index data.

Catch-per-unit-effort (CPUE) and Survey Abundance Observations:

Fleet/ Survey	Units	Error Distribution	SD Report
1	1	0	0
2	1	0	0
...	...	...	...

Units

The options for units for input data are:

- 0 = numbers;

- 1 = biomass;

- 2 = F; and

    - Note the F option can only be used for a fishing fleet and not for a survey, even if the survey selectivity is mirrored to a fishing fleet. The values of these effort data are interpreted as proportional to the level of the fishery F values. No adjustment is made for differentiating between continuous F values versus exploitation rate values coming from Pope's approximation. A normal error structure is recommended so that the input effort data are compared directly to the model's calculated F, rather than to log(F). The resultant proportionality constant has units of $1/Q$ where Q is the catchability coefficient. For more information see the section on Predator effort.

- >=30 Special survey types. These options bypass the calculation of survey selectivity so the no selectivity parameter are required and age/length selectivity pattern should be set as 0. A catchability parameter line in the control file will be required for each special survey. Special survey types 31, 32, and 36 relate to recruitment deviations. Before v.3.30.22, the expected value for observations before recdev_start or after recdev_end were null. With v.3.30.22, expected values are now based on recruitment deviations for all years and suggestions are included in warnings.sso if observations

occur outside the range of active recruitment deviations. The expected values for these types are:

- 30 = spawning biomass/output (e.g., for an egg and larvae survey);

- 31 = exp(recruitment deviation), useful for environmental index affecting recruitment;

- 32 = spawning biomass * exp(recruitment deviation), for a pre-recruit survey occurring before density-dependence;

- 33 = recruitment, age-0 recruits;

- 34 = depletion (spawning biomass/virgin spawning biomass);

  * Special survey option 34 automatically adjusts phases of parameters. To use the depletion survey approach, the user will need to make the following revisions to the SS3 data file: 1) add a new survey fleet, 2) define the survey type as option 34, 3) add two depletion survey data points, and initial unfished set equal to 1 for an unfished modeled year and one for a later year with the depletion estimates, 4) set the input CV value for each survey data point to a low value (e.g., 0.0001) to force the model to fit these data, and in the control file 5) add the survey to the control file in the Q set-up and selectivity sections with float set to 0 with parameter value set to 0.

  * There are options for additional control over this in the control file catchability setup section under the link information bullet where:

    · 0 = add 1 to phases of all parameters. Only R0 active in new phase 1. Mimics the default option of previous model versions;

    · 1 = only R0 active in phase 1. Then finish with no other parameters becoming active; useful for data-limited draws of other fixed parameters. Essentially, this option allows SS3 to mimic DB-SRA; and

    · 2 = no phase adjustments, can be used when profiling on fixed R0.

  * Warning: the depletion survey approach has not been tested on multiple area models. This approach may present challenges depending upon the dynamics within each area.

- 35 = survey of a deviation vector $(e(survey(y)) = f(parm\_dev(k, y)))$, can be used for an environmental time-series with soft linkage to the index. The selected deviation vector is specified in Q section of the control file. The index

of the deviation vector to which the index is related is specified in the 2nd column of the Q setup table (see Catchability);

- 36 = recruitment deviation

Error Distribution

The options for error distribution form are:

- -1 = normal error;

- 0 = lognormal error; and

- >0 = Student's t-distribution in log space with degrees of freedom equal to this value. For DF>30, results will be nearly identical to that for lognormal distribution. A DF value of about 4 gives a fat-tail to the distribution. The standard error values entered in the data file must be the standard error in $\log_e$ space.

Abundance indices typically assumed to have a lognormal error structure with units of standard error of $\log_e$(index). If the variance of the observations is available only as a coefficient of variation (CV = standard error of the observation divided by the mean value of the observation in natural space), then the value of standard error in log space can be calculated as $\sqrt{(log_e(1 + (CV)^2))}$.

For the normal error structure, the entered values for standard error are interpreted directly as a standard error in arithmetic space and not as a CV. Thus switching from a lognormal to a normal error structure forces the user to provide different values for the standard error input in the data file.

If the data exist as a set of normalized Z-scores, you can assert a lognormal error structure after entering the data as exp(Z-score) because it will be logged by SS3. Preferably, the Z-scores would be entered directly and the normal error structure would be used.

Enable SD Report

Indices with SD Report enabled will have the expected values for their historical values appear in the ss.std and ss.cor files. The default value is for this option is 0.

- 0 = SD Report not enabled for this index; and

- 1 = SD Report enabled for this index.

Data Format

Year	Month	Fleet/Survey	Observation	SE
1991	7	3	80000	0.056
1995	7.2	3	65000	0.056
...	...	...	...	...
2000	7.1	3	42000	0.056
-9999	0	0	0	0

- For fishing fleets, CPUE is defined in terms of retained catch (biomass or numbers).

- For fishery independent surveys, retention/discard is not defined so CPUE is implicitly in terms of total CPUE.

- If a survey has its selectivity mirrored to that of a fishery, only the selectivity is mirrored so the expected CPUE for this mirrored survey does not use the retention curve (if any) for the fishing fleet.

- If the fishery or survey has time-varying selectivity, then this changing selectivity will be taken into account when calculating expected values for the CPUE or survey index.

- Year values that are before start year or after end year are excluded from model, so the easiest way to include provisional data in a data file is to put a negative sign on its year value.

- Duplicate survey observations for the same year are not allowed.

- Observations that are to be included in the model but not included in the negative log likelihood need to have a negative sign on their fleet ID. Previously the code for not using observations was to enter the observation itself as a negative value. However, that old approach prevented use of a Z-score environmental index as a "survey". This approach is best for single or select years from an index rather than an approach to remove a whole index. Removing a whole index from the model should be done through the use of lambdas at the bottom of the control file which will eliminate the index from model fitting.

- Observations can be entered in any order, except if the super-year feature is used.

- Super-periods are turned on and then turned back off again by putting a negative sign on the season. Previously, super-periods were started and stopped by entering -9999 and the -9998 in the SE field. See the Data Super-Period section of this manual for more information.

- If the statistical analysis used to create the CPUE index of a fishery has been conducted in such a way that its inherent size/age selectivity differs from the size/age selectivity estimated from the fishery's size and age composition, then you may want to enter the CPUE as if it was a separate survey and with a selectivity that differs from the fishery's estimated selectivity. The need for this split arises because the fishery size and age composition should be derived through a catch-weighted approach (to appropriately represent the removals by the fishery) and the CPUE should be derived through an area-weighted approach to better serve as a survey of stock abundance.

Discard

If discard is not a feature of the model specification, then just a single input is needed:

0	Number of fleets with discard observations

If discard is being used, the input syntax is:

1	Number of fleets with discard observations			
Fleet	Units	Error Distribution		
1	2	-1		
Year	Month	Fleet	Observation	Standard Error
1980	7	1	0.05	0.25
1991	7	1	0.10	0.25
-9999	0	0	0	0

Note that although the user must specify a month for the observed discard data, the unit for discard data is in terms of a season rather than a specific month. So, if using a seasonal model, the input month values must corresponding to some time during the correct season. The actual value will not matter because the discard amount is calculated for the entirety of the season. However, discard length or age observations will be treated by entered observation month.

Discard Units
The options are:

- 1 = values are amount of discard in either biomass or numbers according to the selection made for retained catch;

- 2 = values are fraction (in biomass or numbers) of total catch discarded, biomass/number selection matches that of retained catch; and

- 3 = values are in numbers (thousands) of fish discarded, even if retained catch has units of biomass.

Discard Error Distribution
The four options for discard error are:

- >0 = degrees of freedom for Student's t-distribution used to scale mean body weight deviations. Value of error in data file is interpreted as CV of the observation;

- 0 = normal distribution, value of error in data file is interpreted as CV of the observation;

- -1 = normal distribution, value of error in data file is interpreted as standard error of the observation;

- -2 = lognormal distribution, value of error in data file is interpreted as standard error of the observation in log space; and

- -3 = truncated normal distribution (new with v.3.30, needs further testing), value of error in data file is interpreted as standard error of the observation. This is a good option for low observed discard rates.

Discard Notes

- Year values that are before start year or after end year are excluded from model, so the easiest way to include provisional data in a data file is to put a negative sign on its year value.

- Negative value for fleet causes it to be included in the calculation of expected values, but excluded from the log likelihood.

- Zero (0.0) is a legitimate discard observation, unless lognormal error structure is used.

- Duplicate discard observations from a fleet for the same year are not allowed.

- Observations can be entered in any order, except if the super-period feature is used.

- Note that in the control file you will enter information for retention such that 1-retention is the amount discarded. All discard is assumed dead, unless you enter information for discard mortality. Retention and discard mortality can be either size-based or age-based (new with v.3.30).

Cautionary Note

The use of CV as the measure of variance can cause a small discard value to appear to be overly precise, even with the minimum standard error of the discard observation set to 0.001. In the control file, there is an option to add an extra amount of variance. This amount is added to the standard error, not to the CV, to help correct this problem of underestimated variance.

Mean Body Weight or Length

This is the overall mean body weight or length across all selected sizes and ages. This may be useful in situations where individual fish are not measured but mean weight is obtained by counting the number of fish in a specified sample (e.g., a 25 kg basket).

Mean Body Weight Data Section:

1	Use mean body size data (0/1)

COND > 0:

30	Degrees of freedom for Student's t-distribution used to evaluate mean body weight deviation.

Year	Month	Fleet	Partition	Type	Observation	CV
1990	7	1	0	1	4.0	0.95
1990	7	1	0	1	1.0	0.95
-9999	0	0	0	0	0	0

Partition

Mean weight data and composition data require specification of what group the sample originated from (e.g., discard, retained, discard + retained). Note: if retention is not defined in the selectivity section, observations with Partition = 2 will be changed to Partition = 0.

- 0 = combined catch in units of weight (whole, e.g. discard + retained);

- 1 = discarded catch in units of weight; and

- 2 = retained catch in units of weight.

Type

Specify the type of data:

- 1 = mean length; and

- 2 = mean body weight.

Observation - Units

Units must correspond to the units of body weight, normally in kilograms, (or mean length in cm). The expected value of mean body weight (or mean length) is calculated in a way that incorporates effect of selectivity and retention.

Error

Error is entered as the CV of the observed mean body weight (or mean length)

Population Length Bins

The first part of the length composition section sets up the bin structure for the population. These bins define the granularity of the age-length key and the coarseness of the length selectivity. Fine bins create smoother distributions, but a larger and slower running model. First read a single value to select one of three population length bin methods, then any conditional input for options 2 and 3:

1	Use data bins to be read later. No additional input here.
2	generate from bin width min max, read next:
	2 Bin width
	10 Lower size of first bin
	82 Lower size of largest bin
The number of bins is then calculated from: (max Lread - min Lread)/(bin width) + 1	
3	Read 1 value for number of bins, and then read vector of bin boundaries
	37 Number of population length bins to be read
	10 12 14 ... 82 Vector containing lower edge of each population size bin

Notes

There are some items for users to consider when setting up population length bins:

- For option 2, bin width should be a factor of min size and max size. For options 2 and 3, the data length bins must not be wider than the population length bins and the boundaries of the bins do not have to align. The transition matrix between population and data length bins is output to echoinput.sso.

- The mean size at settlement (virtual recruitment age) is set equal to the min size of the first population length bin.

- When using more, finer population length bins, the model will create smoother length selectivity curves and smoother length distributions in the age-length key, but run more slowly (more calculations to do).

- The mean weight-at-length, maturity-at-length and size-selectivity are based on the mid-length of the population bins. So these quantities will be rougher approximations if broad bins are defined.

- Provide a wide enough range of population size bins so that the mean body weight-at-age will be calculated correctly for the youngest and oldest fish. If the growth curve extends beyond the largest size bin, then these fish will be assigned a length equal to the mid-bin size for the purpose of calculating their body weight.

- While exploring the performance of models with finer bin structure, a potentially pathological situation has been identified. When the bin structure is coarse (note that some applications have used 10 cm bin widths for the largest fish), it is possible for a selectivity slope parameter or a retention parameter to become so steep that all of the action occurs within the range of a single size bin. In this case, the model will see zero gradient of the log likelihood with respect to that parameter and convergence will be hampered.

- A value read near the end of the starter.ss file defines the degree of tail compression used for the age-length key, called ALK tolerance. If this is set to 0.0, then no compression is used and all cells of the age-length key are processed, even though they may contain trivial (e.g., 1 e-13) fraction of the fish at a given age. With tail compression of, say 0.0001, the model, at the beginning of each phase, will calculate the min and max length bin to process for each age of each morphs ALK and compress accordingly. Depending on how many extra bins are outside this range, you may see speed increases near 10-20%. Large values of ALK tolerance, say 0.1, will create a sharp end to each distribution and likely will impede convergence. It is recommended to start with a value of 0 and if model speed is an issue, explore values greater than 0 and evaluate the trade-off between model estimates and run time. The user is encouraged to explore this feature.

Length Composition Data Structure

Enter a code to indicate whether or not length composition data will be used:

1	Use length composition data (0/1/2)

If the value 0 is entered, then skip all length related inputs below and skip to the age data setup section. If value 1 is entered, all data weighting options for composition data apply equally to all partitions within a fleet. If value 2 is entered, then the data weighting options are applied by the partition specified. Note that the partitions must be entered in numerical order within each fleet.

If the value for fleet is negative, then the vector of inputs is copied to all partitions (0 = combined, 1 = discard, and 2 = retained) for that fleet and all higher numbered fleets. This as a good practice so that the user controls the values used for all fleets.

Example table of length composition settings when "Use length composition data" = 1 (where here the first fleet has multinomial error structure with no associated parameter, and the second fleet uses Dirichlet-multinomial structure):

Min. Tail Compress.	Constant added to prop.	Combine males & females	Compress. Bins	Comp. Error Dist.	Param. Select	Min. Sample Size
0	0.0001	0	0	0	0	0.1
0	0.0001	0	0	1	1	0.1

Example table of length composition settings when "Use length composition data" = 2 (where here the -1 in the fleet column applies the first parameter to all partitions for fleet 1 while fleet 2 has separate parameters for discards and retained fish):

Fleet	Partition	Min. Tail Compress.	Constant added to prop.	Combine males & females	Compress. Bins	Comp. Error Dist.	Param. Select	Min. Sample Size
-1	0	0	0.0001	0	0	1	1	0.1
2	1	0	0.0001	0	0	1	2	0.1
2	2	0	0.0001	0	0	1	3	0.1
...								
-9999	0	0	0	0	0	0	0	0

Minimum Tail Compression

Compress tails of composition until observed proportion is greater than this value; negative value causes no compression; Advise using no compression if data are very sparse, and especially if the set-up is using age composition within length bins because of the sparseness of these data. A single fish being observed with tail compression on will cause the entire vector to be collapsed to that bin.

Added Constant to Proportions

Constant added to observed and expected proportions at length and age to make logL calculations more robust. Tail compression occurs before adding this constant. Proportions are renormalized to sum to 1.0 after constant is added.

The constant should be greater than 0. Commonly used values range from 0.00001 to 0.01. Larger values will cause differences among bins with smaller values to be less influential, leading to greater relative influence of the bins with largest proportions of the compositions.

Combine Males & Females

Combine males into females at or below this bin number. This is useful if the sex determination of very small fish is doubtful so allows the small fish to be treated as combined sex. If Combine Males & Females > 0, then add males into females for bins 1 through this number, zero out the males, set male data to start at the first bin above this bin. Note that Combine Males & Females > 0 is entered as a bin index, not as the size associated with that bin. Comparable option is available for age composition data.

Compress Bins

This option allows for the compression of length or age bins beyond a specific length or age by each data source. As an example, a value of 5 in the compress bins column would condense the final five length bins for the specified data source.

Composition Error Distribution

The options are:

- 0 = Multinomial Error;

- 1 = Dirichlet Multinomial Error (linear); and

    - The Dirichlet Multinomial Error distribution requires the addition of a parameter lines for the natural log of the effective sample size multiplier (θ) at the end of the selectivity parameter section in the control file. See the Dirichlet parameter in the control file for information regarding setup.

    - The Parameter Select option needs be used to specify which data sources should be weighted together or separate.

- 2 = Dirichlet Multinomial Error (saturation).

    - This parameterization of the Dirichlet-multinomial Error has not been tested, so this option should be used with caution. The Dirichlet Multinomial Error data weighting approach will calculate the effective sample size based on equation 12

58

from Thorson et al. (2017) where the estimated parameter will now be in terms of β. The application of this method should follow the same steps detailed above for option 1.

Parameter Select

Value that indicates the groups of composition data for estimation of the Dirichlet parameter for weighting composition data.

- 0 = Default; and

- 1-N = Only used for the Dirichlet option. Set to a sequence of numbers from 1 to N where N is the total number of combinations of fleet and age/length. That is, if you have 3 fleets with length data, but only 2 also have age data, you would have values 1 to 3 in the length comp setup and 4 to 5 in the age comp setup. You can also have a data weight that is shared across fleets by repeating values in Parameter Select. Note that there can be no skipped numbers in the sequence from 1 to N, otherwise the model will exit on error when reading in the input files.

Minimum Sample Size

The minimum value (floor) for all sample sizes. This value must be at least 0.001. Conditional age-at-length data may have observations with sample sizes less than 1. Version 3.24 had an implicit minimum sample size value of 1.

Additional information on Dirichlet Parameter Number and Effective Sample Sizes

If the Dirichlet-multinomial error distribution is selected, indicate here which of a list of Dirichlet-multinomial parameters will be used for this fleet. So each fleet could use a unique Dirichlet-multinomial parameter, or all could share the same, or any combination of unique and shared. The requested number of Dirichlet-multinomial parameters are specified as parameter lines in the control file immediately after the selectivity parameter section. Please note that age-compositions Dirichlet-multinomial parameters are continued after length-compositions, so a model with one fleet and both data types would presumably require two new Dirichlet-multinomial parameters.

The Dirichlet estimates the effective sample size as $N_{eff} = \frac{1}{1+\theta} + \frac{N\theta}{1+\theta}$ where θ is the estimated parameter and N is the input sample size. Stock Synthesis estimates the log of the Dirichlet-multinomial parameter such that $\hat{\theta}_{\text{fishery}} = e^{-0.6072} = 0.54$ where assuming $N = 100$ for the fishery would result in an effective sample size equal to 35.7.

This formula for effective sample size implies that, as the Stock Synthesis parameter ln(DM_theta) goes to large values (i.e., 20), then the adjusted sample size will converge to the input sample size. In this case, small changes in the value of the ln(DM_theta) parameter has no action, and the derivative of the negative log-likelihood is zero with

respect to the parameter, which means the Hessian will be singular and cannot be inverted. To avoid this non-invertible Hessian when the ln(DM_theta) parameter becomes large, turn it off while fixing it at the high value. This is equivalent to turning off down-weighting of fleets where evidence suggests that the input sample sizes are reasonable.

For additional information about the Dirichlet-multinomial please see Thorson et al. (2017) and the detailed Data Weighting section.

Length Composition Data

Composition data can be entered as proportions, numbers, or values of observations by length bin based on data expansions.

The data bins do not need to cover all observed lengths. The selection of data bin structure should be based on the observed distribution of lengths and the assumed growth curve. If growth asymptotes at larger lengths, having additional length bins across these sizes may not contribute information to the model and may slow model run time. Additionally, the lower length bin selection should be selected such that, depending on the size selection, to allow for information on smaller fish and possible patterns in recruitment. While set separately users should ensure that the length and age bins align. It is recommended to explore multiple configurations of length and age bins to determine the impact of this choice on model estimation.

Specify the length composition data as:

28	Number of length bins for data
26 28 30 … 80	Vector of length bins associated with the length data

Note: the vector of length bins above will aggregate data from outside the range of values as follows:

	bin 1	bin 2	bin 3	…	bin 27	bin 28
bin vector	26	28	30	…	78	80
bin contains	0–27.99	28–29.99	30–30.99	…	78–79.99	80+

Example of a single length composition observation:

Year	Month	Fleet	Sex	Partition	Nsamp	data vector
1986	1	1	3	0	20	\<female then male data\>
...	...	...	...	...	...	...
-9999	0	0	0	0	0	\<0 repeated for each element of the data vector above\>

Sex

If model has only one sex defined in the set-up, all observations must have sex set equal to 0 or 1 and the data vector by year will equal the number of the user defined data bins. This also applies to the age data.

In a 2 sex model, the data vector always has female data followed by male data, even if only one of the two sexes has data that will be used. The below description applies to a 2 sex model:

- Sex = 0 means combined male and female (must already be combined and information placed in the female portion of the data vector) (male entries must exist for correct data reading, then will be ignored).

- Sex = 1 means female only (male entries must exist for correct data reading, then will be ignored).

- Sex = 2 means male only (female entries must exist and will be ignored after being read).

- Sex = 3 means data from both sexes will be used and they are scaled so that they together sum to 1.0; i.e. sex ratio is preserved.

Partition

Partition indicates samples from either combined, discards, or retained catch. Note: if retention is not defined in the selectivity section, observations with Partition = 2 will be changed to Partition = 0.

- 0 = combined (whole, e.g. discard + retained);

- 1 = discard; and

- 2 = retained.

Excluding Data

- If the value of year is negative, then that observation is not transferred into the working array. This feature is the easiest way to include observations in a data file but not to use them in a particular model scenario.

- If the value of fleet in the length or age composition observed data line is negative, then the observation is processed and its expected value and log likelihood are calculated, but this log likelihood is not included in the total log likelihood. This feature allows the user to see the fit to a provisional observation without having that observation affect the model.

Note

When processing data to be input into SS3, all observed fish of sizes smaller than the first bin should be added to the first bin and all observed fish larger than the last bin should be condensed into the last bin.

The number of length composition data lines no longer needs to be specified in order to read the length (or age) composition data. Starting in v.3.30, the model will continue to read length composition data until an pre-specified exit line is read. The exit line is specified by entering -9999 at the end of the data matrix. The -9999 indicates to the model the end of length composition lines to be read.

Each observation can be stored as one row for ease of data management in a spreadsheet and for sorting of the observations. However, the 6 header values, the female vector and the male vector could each be on a separate line because ADMB reads values consecutively from the input file and will move to the next line as necessary to read additional values.

The composition observations can be in any order and replicate observations by a year for a fleet are allowed (unlike survey and discard data). However, if the super-period approach is used, then each super-periods' observations must be contiguous in the data file.

Age Composition Option

The age composition section begins by reading the number of age bins. If the value 0 is entered for the number of age bins, then skips reading the bin structure and all reading of other age composition data inputs.

17	Number of age bins; can be equal to 0 if age data are not used; do not include a vector of agebins if the number of age bins is set equal to 0.

Age Composition Bins

If a positive number of age bins is read, then reads the bin definition next.

1 2 3 ... 20 25	Vector of ages

The bins are in terms of observed age (here age) and entered as the lower edge of each bin. Each ageing imprecision definition is used to create a matrix that translates true age structure into age structure. The first and last age' bins work as accumulators. So in the example any age 0 fish that are caught would be assigned to the age = 1 bin.

Ageing Error

Here, the capability to create a distribution of age (e.g., age with possible bias and imprecision) from true age is created. One or many ageing error definitions can be created. For each, the model will expect an input vector of mean age and a vector of standard deviations associated with the mean age.

2	Number of ageing error matrices to generate

Example with no bias and very little uncertainty at age Tstrut

Age-0	Age-1	Age-2	...	Max Age	
-1	-1	-1	...	-1	#Mean Age
0.001	0.001	0.001	...	0.001	#SD

Example with no bias and some uncertainty at age:

0.5	1.5	2.5	...	Max Age + 0.5	#Mean Age
0.5	0.65	0.67	...	4.3	#SD Age

Example with bias and uncertainty at age:

0.5	1.4	2.3	...	Max Age + Age Bias	#Mean Age
0.5	0.65	0.67	...	4.3	#SD Age

In principle, one could have year or laboratory specific matrices for ageing error. For each matrix, enter a vector with mean age for each true age; if there is no ageing bias, then set age equal to true age + 0.5. Alternatively, -1 value for mean age means to set it equal to

true age plus 0.5. The addition of +0.5 is needed so that fish will get assigned to the intended integer age. The length of the input vector is equal to the population maximum age plus one (0-max age), with the first entry being for age 0 fish and the last for fish of population maximum age even if the maximum age bin for the data is lower than the population maximum age. The following line is a a vector with the standard deviation of age for each true age with a normal distribution assumption.

The model is able to create one ageing error matrix from parameters, rather than from an input vector. The range of conditions in which this new feature will perform well has not been evaluated, so it should be considered as a preliminary implementation and subject to modification. To invoke this option, for the selected ageing error vector, set the standard deviation of ageing error to a negative value for age 0. This will cause creation of an ageing error matrix from parameters and any age or size-at-age data that specify use of this age error pattern will use this matrix. Then in the control file, add a full parameter line below the cohort growth deviation parameter (or the movement parameter lines if used) in the mortality growth parameter section. These parameters are described in the control file section of this manual.

Code for ageing error calculation can be found in SS_miscfxn.tpl, search for function "get_age_age" or "SS_Label_Function 45".

Age Composition Specification

If age data are included in the model, the following set-up is required, similar to the length data section. See Length Composition Data Structure for details on each of these inputs.

Specify bin compression and error structure for age composition data for each fleet:

Min. Tail Compress.	Constant added to prop.	Combine males & females	Compress. Bins	Comp. Error Dist.	Param. Select	Min. Sample Size
0	0.0001	1	0	0	0	1
0	0.0001	1	0	0	0	1

Specify method by which length bin range for age obs will be interpreted:

1	Bin method for age data
	1 = value refers to population bin index
	2 = value refers to data bin index
	3 = value is actual length (which must correspond to population length bin boundary)

An example age composition observation:

Year	Month	Fleet	Sex	Partition	Age Err	Lbin lo	Lbin hi	Nsamp	Data Vector
1987	1	1	3	0	2	-1	-1	79	<enter data values>
-9999	0	0	0	0	0	0	0	0	0

Syntax for Sex, Partition, and data vector are same as for length. The data vector has female values then male values, just as for the length composition data.

Age Error

Age error (Age Err) identifies which ageing error matrix to use to generate expected value for this observation.

Lbin Low and Lbin High

Lbin lo and Lbin hi are the range of length bins that this age composition observation refers to. Normally these are entered with a value of -1 and -1 to select the full size range. Whether these are entered as population bin number, length data bin number, or actual length is controlled by the value of the length bin range method above.

- Entering value of 0 or -1 for Lbin lo converts Lbin lo to 1;

- Entering value of 0 or -1 for Lbin hi converts Lbin hi to Maxbin;

- It is strongly advised to use the -1 codes to select the full size range. If you use explicit values, then the model could unintentionally exclude information from some size range if the population bin structure is changed.

- In reporting to the comp_report.sso, the reported Lbin_lo and Lbin_hi values are always converted to actual length.

Excluding Data

As with the length composition data, a negative year value causes the observation to not be read into the working matrix, a negative value for fleet causes the observation to be included in expected values calculation, but not in contribution to total log likelihood, a negative value for month causes start-stop of super-period.

Conditional Age-at-Length

Use of conditional age-at-length will greatly increase the total number of age composition observations and associated model run time but there can be several advantages to inputting ages in this fashion. First, it avoids double use of fish for both age and size information because the age information is considered conditional on the length information. Second, it contains more detailed information about the relationship between size and age so provides stronger ability to estimate growth parameters, especially the variance of size-at-age. Lastly, where age data are collected in a length-stratified program, the conditional age-at-length approach can directly match the protocols of the sampling program.

However, simulation research has shown that the use of conditional age-at-length data can result in biased growth estimates in the presence of unaccounted for age-based movement when length-based selectivity is assumed (Lee et al., 2017), when other age-based processes (e.g., mortality) are not accounted for (Lee et al., 2019), or based on the age sampling protocol (Piner et al., 2016). Understanding how data are collected (e.g., random, length-conditioned samples) and the biology of the stock is important when using conditional age-at-length data for a fleet.

In a two sex model, it is best to enter these conditional age-at-length data as single sex observations (sex = 1 for females and = 2 for males), rather than as joint sex observations (sex = 3). Inputting joint sex observations comes with a more rigid assumption about sex ratios within each length bin. Using separate vectors for each sex allows 100% of the expected composition to be fit to 100% observations within each sex, whereas with the sex = 3 option, you would have a bad fit if the sex ratio were out of balance with the model expectation, even if the observed proportion at age within each sex exactly matched the model expectation for that age. Additionally, inputting the conditional age-at-length data as single sex observations isolates the age composition data from any sex selectivity as well.

Conditional age-at-length data are entered within the age composition data section and can be mixed with marginal age observations for other fleets of other years within a fleet. To treat age data as conditional on length, Lbin_lo and Lbin_hi are used to select a subset of the total size range. This is different than setting Lbin_lo and Lbin_hi both to

-1 to select the entire size range, which treats the data entered on this line within the age composition data section as marginal age composition data.

An example conditional age-at-length composition observations:

Year	Month	Fleet	Sex	Partition	Age Err	Lbin lo	Lbin hi	Nsamp	Data Vector
1987	1	1	1	0	2	10	10	18	<data values>
1987	1	1	1	0	2	12	12	24	<data values>
1987	1	1	1	0	2	14	14	16	<data values>
1987	1	1	1	0	2	16	16	30	<data values>
-9999	0	0	0	0	0	0	0	0	0

In this example observation, the age data is treated as on being conditional on the 2 cm length bins of 10–11.99, 12–13.99, 14–15.99, and 16–17.99cm. If there are no observations of ages for a specific sex within a length bin for a specific year, that entry may be omitted.

Mean Length or Body Weight-at-Age

The model also accepts input of mean length-at-age or mean body weight-at-age. This is done in terms of observed age, not true age, to take into account the effects of ageing imprecision on expected mean size-at-age. If the value of the Age Error column is positive, then the observation is interpreted as mean length-at-age. If the value of the Age Error column is negative, then the observation is interpreted as mean body weight-at-age and the abs(Age Error) is used as Age Error.

1			Use mean size-at-age observation (0 = none, 1 = read data matrix)					

An example observation:

Yr	Month	Fleet	Sex	Part.	Age Err.	Ignore	Data Vector (Female - Male)	Sample Size (Female - Male)
1989	7	1	3	0	1	999	<Mean Size values>	<Sample Sizes>
...								
-9999	0	0	0	0	0	0	0 0 0 0 0 0 0	0 0 0 0 0 0 0

Note

- Negatively valued mean size entries with be ignored in fitting. This feature allows the user to see the fit to a provisional observation without having that observation affect the model.

- A number of fish value of 0 will cause mean size value to be ignored in fitting. If the number of fish is zero, a non-zero mean size or body weight-at-age value, such as 0.01 or -999, still needs to be added. This feature allows the user to see the fit to a provisional observation without having that observation affect the model.

- Negative value for year causes observation to not be included in the working matrix. This feature is the easiest way to include observations in a data file but not to use them in a particular model scenario.

- Each sexes' data vector and N fish vector has length equal to the number of age bins.

- The "Ignore" column is not used (set aside for future options) but still needs to have default values in that column (any value).

- Where age data are being entered as conditional age-at-length and growth parameters are being estimated, it may be useful to include a mean length-at-age vector with nil emphasis to provide another view on the model's estimates.

- An experiment that may be of interest might be to take the body weight-at-age data and enter it to the model as empirical body weight-at-true age in the wtatage.ss file, and to contrast results to entering the same body weight-at-age data here and to attempt to estimate growth parameters, potentially time-varying, that match these body weight data.

- If using mean size-at-age data, please see the lambda usage notes regarding issues for model fitting depending upon other data within the model.

Environmental Data

The model accepts input of time series of environmental data. Parameters can be made to be time-varying by making them a function of one of these environmental time series. In v.3.30.16 the option to specify the centering of environmental data by either using the mean of the by mean and the z-score.

Parameter values can be a function of an environmental data series:

1	Number of environmental variables

The environmental data can be centered by subtracting the mean and dividing by stdev (z-score, -1) or by subtracting the mean of the environmental variable (-2) based on the year column value.

COND > 0 Example of 2 environmental observations:

Year	Variable	Value
1990	1	0.10
1991	1	0.15
-1	1	1
-2	2	1
-9999	0	0

The final two lines in the example above indicate in that variable series 1 will be centered by subtracting the mean and dividing by the standard deviation (indicated by the -1 value in the year column). The environmental variable series 2 will be centered by subtracting the mean of the time series (indicated by the -2 value in the year column). The input in the "value" column for both of the final two lines specifying the centering of the time series is ignored by the model. The control file also will need to be modified to in the long parameter line column "env-var" for the selected parameter. This feature was added in v.3.30.16.

Note

- Any years for which environmental data are not read are assigned a value of 0.0. None of the current link functions contain a link parameter that acts as an offset. Therefore, you should subtract the mean from your data. This lessens the problem with missing observations, but does not eliminate it. A better approach for dealing with missing observations is to use a different approach for the environmental effect on the parameter. Set up the parameter to have random deviations for all years, then enter the zero-centered environmental information as a special survey of type 35 and set up the catchability of that survey to be a link to the deviation vector. This is a more complex approach, but it is superior in treatment of missing values and superior in allowing for error in the environmental relationship.

- Users can assign environmental conditions for the initial equilibrium year by including environmental data for one year before the start year. However, this works only for recruitment parameters, not biology or selectivity parameters.

- Environmental data can be read for up to 100 years after the end year of the model. Then, if the recruitment-environment link has been activated, the future recruitments will be influenced by any future environmental data. This could be used to create a future "regime shift" by setting historical values of the relevant environmental variable equal to zero and future values equal to 1, in which case the magnitude of the regime shift would be dictated by the value of the environmental linkage parameter. Note that only future recruitment and growth can be modified by the environmental inputs; there are no options to allow environmentally-linked selectivity in the forecast years.

Generalized Size Composition Data

The generalized approach to size composition information was designed initially to provide a means to include weight frequency data. However, the uses are broader, such as allowing for size composition data with different data bins. The user can define as many generalized size composition methods as necessary.

- Each method has a specified number of bins.

- Each method has "units" so the frequencies can be in units of biomass or numbers.

- Each method has "scale" so the bins can be in terms of weight or length (including ability to convert bin definitions in pounds or inches to kg or cm).

- The composition data is input as females then males, just like all other composition data in SS3. In a two-sex model, the new composition data can be combined sex, single sex, or both sex.

- The generalized size composition data can be from the combined discard and retained (i.e., whole), discard only, or retained only.

- There are two options for treating fish that in population size bins are smaller than the smallest size frequency bin.

    - Option 1: By default, these fish are excluded (unlike length composition data where the small fish are automatically accumulated up into the first bin).

    - Option 2: If the first size bin is given as a negative value, then accumulation is turned on and the absolute value of the entered value is used as the lower edge of the first size bin.

Example entry:

2	Number (N) of size frequency methods to be read. If this value is 0, then omit all entries below. A value of -1 (or any negative value) triggers expanded optional inputs below that allow for Dirichlet for fitting these data.

COND < 0 - Number of size frequency

2	Number of size frequency methods to read

END COND < 0

25 15	Number of bins per method
2 2	Units per each method (1 = biomass, 2 = numbers)
3 3	Scale per each method (1 = kg, 2 = lbs, 3 = cm, 4 = inches)
1e-9 1e-9	Min compression to add to each observation (entry for each method)
2 2	Number of observations per weight frequency method

COND < 0 - Number of size frequency

1 1	Composition error structure (0 = multinomial, 1 = Dirichlet using Theta*n, 2 = Dirichlet using beta)
1 1	Parameter select consecutive index for Dirichlet composition error

END COND < 0

Then enter the lower edge of the bins for each method. The two row vectors shown below contain the bin definitions for methods 1 and 2 respectively:

-26	28	30	32	34	36	38	40	42	...	60	62	64	68	72	76	80	90

-26	28	30	32	34	36	38	40	42	44	46	48	50	52	54

Example input is shown below. Note that the format is identical to the length composition data, including sex and partition options, except for the addition of the first column, which indicates the size frequency method.

Method	Year	Month	Fleet	Sex	Part	Sample Size	<composition females then males>
1	1975	1	1	3	0	43	<data>
1	1977	1	1	3	0	43	<data>
1	1979	1	1	3	0	43	<data>
1	1980	1	1	3	0	43	<data>

Note

- There is no tail compression for generalized size frequency data.

- Super-period capability is as for the length and age composition data.

- By choosing units = 2 and scale = 3 with identical bins and a negative first bin to turn accumulation of small fish on, the size composition method is identical to the length composition method.

- Bin boundaries do not need to align with the population length bin boundaries. The model interpolates as necessary.

- Size bins cannot be defined as narrower than the population bin width.

- The transition matrix can depend upon weight-at-length which differs between sexes and can vary seasonally. Thus, the transition matrix is calculated internally for each sex and each season.

Tag-Recapture Data

Each released tag group is characterized by an area, time, sex and age at release. Each recapture event is characterized by a time and fleet (since fleets operate in only one area, it is not necessary to specify the area of recapture). Fleets with tagging data must be fishing fleets (e.g., fleet type 1 or 2).

Inside the model, the tagged cohort is apportioned across all growth patterns in a given area at a given time (with options to apportion to only one sex or to both). The tag cohort by growth pattern then behaves according to the movement and mortality of the growth pattern. The number of tagged fish is modeled as a negligible fraction of the total population, so a tagging event does not move fish from an untagged group to a tagged group. Instead, tagged fish are seeded into the population with no impact at all on the total population abundance or mortality.

Predominant age at release for each tag group must be assigned; this requirement keeps SS3 efficient. By assigning a tag group to a single age rather than distributing it across all possible ages according to the size composition of the release group, the tag group can be tracked as a single cohort through the age by time matrix with minimal overhead to the rest of the model. Tags are released at the beginning of a season and recaptures follow the timing of the fleet that made the recapture.

Example set-up for tagging data:

1	Do tags - 0/1/2. If this value is 0, then omit all entries below.
	If value is 2, read 1 additional input.

COND > 0	All subsequent tag-recapture entries must be omitted if "Do Tags" = 0	
	3	Number of tag groups
	7	Number of recapture events
	2	Mixing latency period: N periods to delay before comparing observed to expected recoveries (0 = release period).
	10	Max periods (seasons) to track recoveries, after which tags enter accumulator

COND = 2		
	2	Minimum recaptures. The number of recaptures >= mixperiod must be >= min tags recaptured specified to include tag group in log likelihood

Release Data

TG	Area	Year	Season	\<tfill\>	Sex	Age	N Release
1	1	1980	1	999	0	24	2000
2	1	1995	1	999	1	24	1000
3	1	1985	1	999	2	24	10

Recapture Data

TG	Year	Season	Fleet	Number
1	1982	1	1	7
1	1982	1	2	5
1	1985	1	2	0
2	1997	1	1	6
2	1997	2	1	4
3	1986	1	1	7
3	1986	2	1	5

Note

- The release data must be entered in tag group order.

- <tfill> values are place holders and are replaced by program generated values for model time.

- Analysis of the tag-recapture data has one negative log likelihood component for the distribution of recaptures across areas and another negative log likelihood component for the decay of tag recaptures from a group over time. Note the decay of tag recaptures from a group over time suggests information about mortality is available in the tag-recapture data. More on this is in the control file documentation.

- Do tags option 2 adds an additional input compared to do tags option 1, minimum recaptures. Minimum recaptures allows the user to exclude tag groups that have few recaptures after the mixing period from the likelihood. This may be useful when few tags from a group have been recaptured as an alternative to manually removing the groups with these low numbers of recaptured tags from the tagging data.

- Warning for earlier versions of SS3: A shortcoming in the recapture calculations when also using Pope's F approach was identified and corrected in v.3.30.14.

Stock (Morph) Composition Data

It is sometimes possible to observe the fraction of a sample that is composed of fish from different stocks. These data could come from genetics, otolith microchemistry, tags, or other means. The growth pattern feature allows definition of cohorts of fish that have different biological characteristics and which are independently tracked as they move among areas. SS3 now incorporates the capability to calculate the expected proportion of a sample of fish that come from different growth patterns, "morphs". In the inaugural application of this feature, there was a 3 area model with one stock spawning and recruiting in area 1, the other stock in area 3, then seasonally the stocks would move into area 2 where stock composition observations were collected, then they moved back to their natal area later in the year.

Stock composition by growth pattern (morph) data can be entered in as follows:

1	Do morph composition, if zero, then do not enter any further input below.

COND = 1

3	Number of observations

2	Number of morphs

0.0001	Minimum Compression

Year	Month	Fleet	Null	Nsamp	Data by N Morphs	
1980	1	1	0	36	0.4	0.6
1981	1	1	0	40	0.44	0.54
1982	1	1	0	50	0.37	0.63

Note

- The number of stocks entered with these data must match the number of growth patterns (morphs) in the control file.

- Each data line for unique observations should enter data for morph 1 first followed sequentially for each morph included in the model.

- The expected value is combined across sexes. The entered data values will be normalized to sum to one within SS3.

- The "null" flag is included here in the data input section and is a reserved spot for future features.

- Note that there is a specific value of minimum compression to add to all values of observed and expected.

- Warning for earlier versions of SS3: A flaw was identified in the calculation of accumulation by morph. This has been corrected in v.3.30.14. Older versions were incorrectly calculating the catch by morph using the expectation around age-at-length which already was accounting for the accumulation by morph.

Selectivity Empirical Data (future feature)

It is sometimes possible to conduct field experiments or other studies to provide direct information about the selectivity of a particular length or age relative to the length or age that has peak selectivity, or to have a prior for selectivity that is more easily stated than a

prior on a highly transformed selectivity parameter. This section provides a way to input data that would be compared to the specified derived value for selectivity. This is a placeholder at this time, required to include in the data file and will be fully implemented soon.

Selectivity data feature is under development for a future option and is not yet implemented. The input line still must be specified in as follows:

0	Do data read for selectivity (future option)

End of Data File

999	#End of data file marker

Excluding Data

Data that are before the model start year or greater than the retrospective year are not moved into the internal working arrays at all. So if you have any alternative observations that are used in some model runs and not in others, you can simply give them a negative year value rather than having to comment them out. The first output to data.ss_new has the unaltered and complete input data. Subsequent reports to data.ss_new produce expected values or bootstraps only for the data that are being used. Additional information on bootstrapping is available in Bootstrap Data Files Section.

Data that are to be included in the calculations of expected values, but excluded from the calculation of negative log likelihood, are flagged by use of a negative value for fleet number.

Data Super-Periods

The super-period capability allows the user to introduce data that represent a blend across a set of time steps and to cause the model to create an expected value for this observation that uses the same set of time steps. The option is available for all types of data and a similar syntax is used.

All super-period observations must be contiguous in the data file. All but one of the observations in the sequence will have a negative value for fleet ID so the data associated with these dummy observations will be ignored. The observed values must be combined outside of the model and then inserted into the data file for the one observation with a positive fleet number.

Super-periods are started with a negative value for month, and then stopped with a negative value for month, observations within the super-period are designated with a negative fleet field. The standard error or input sample size field is now used for weighting of the expected values. An error message is generated if the super-period does not contain one observation with a positive fleet field.

An expected value for the observation will be computed for each selected time period within the super-period. The expected values are weighted according to the values entered in the standard error (or input sample size) field for all observations except the single observation holding the combined data. The expected value for that year gets a relative weight of 1.0. So in the example below, the relative weights are: 1982, 1.0 (fixed); 1983, 0.85; 1985, 0.4; 1986, 0.4. These weights are summed and rescaled to sum to 1.0, and are output in the echoinput.sso file.

Not all time steps within the extent of a super-period need be included. For example, in a three season model, a super-period could be set up to combine information from season 2 across 3 years, e.g., skip over the season 1 and season 3 for the purposes of calculating the expected value for the super-period. The key is to create a dummy observation (negative fleet value) for all time steps, except 1, that will be included in the super-period and to include one real observation (positive fleet value; which contains the real combined data from all the specified time steps).

Super-period example:

Year	Month	Fleet	Obs	SE	Comment
1982	-2	3	34.2	0.3	Start super-period. This observation has positive fleet value, so is expected to contain combined data from all identified periods of the super-period. The standard error (SE) entered here is use as the SE of the combined observation. The expected value for the survey in 1982 will have a relative weight of 1.0 (default) in calculating the combined expected value.
1983	2	-3	55	0.3	In super-period; entered observation is ignored. The expected value for the survey in 1983 will have a relative weight equal to the value in the standard error field (0.3) in calculating the combined expected value.
1985	2	-3	88	0.40	Note that 1984 is not included in the super-period. Relative weight for 1985 is 0.4
1986	-2	-3	88	0.40	End super-period

A time step that is within the time extent of the super-period can still have its own separate observation. In the above example, the survey observation in 1984 could be entered as a separate observation, but it must not be entered inside of the contiguous block of super-period observations. For composition data (which allow for replicate observations), a particular time steps' observations could be entered as a member of a super-period and as a separate observation.

The super-period concept can also be used to combine seasons within a year with multiple seasons. This usage could be preferred if fish are growing rapidly within the year so their effective age selectivity is changing within year as they grow; fish are growing within the year so fishery data collected year round have a broader size-at-age modes than a mid-year model approximation can produce; and it could be useful in situations with very high fishing mortality.

Control File

Overview of Control File

These listed model features are denoted in the control file in the following order:

1. Number of growth patterns and platoons

2. Design matrix for assignment of recruitment to area/settlement event/growth pattern

3. Design matrix for movement between areas

4. Definition of time blocks that can be used for time-varying parameters

5. Controls far all time-varying parameters

6. Specification for growth and fecundity

7. Natural mortality growth parameters, weight-at-length, maturity, and fecundity, for each sex

8. Hermaphroditism parameter line (if used)

9. Recruitment distribution parameters for each area, settlement event, and growth pattern

10. Cohort growth deviation

11. Movement between areas (if used)

12. Age error parameter line (if used)

13. Catch multiplier (if used)

14. Fraction female

15. Setup for any mortality-growth parameters are time-varying

16. Seasonal effects on biology parameters

17. Spawner-recruitment parameters

18. Setup for any stock recruitment parameters are time-varying

19. Recruitment deviations

20. F ballpark value in specified year

21. Method for calculating fishing mortality (F)

22. Initial equilibrium F for each fleet

23. Catchability (Q) setup for each fleet and survey

24. Catchability parameters

25. Setup for any catchability parameters are time-varying

26. Length selectivity, retention, discard mortality setup for each fleet and survey

27. Age selectivity setup for each fleet and survey

28. Parameters for length selectivity, retention, discard mortality for each fleet and survey

29. Parameters for age selectivity, retention, discard mortality for each fleet and survey

30. Setup for any selectivity parameters that are time-varying

31. Tag-recapture parameters

32. Variance adjustments

33. Lambdas for likelihood components

The order in which they appear in the control file has grown over time rather opportunistically, so it may not appear particularly logical at this time, especially various aspects of recruitment distribution and growth. When the same information is entered via the SS3 GUI, it is organized more logically and then written in this form to the text control file.

Parameter Line Elements

The primary role of the control file is to define the parameters to be used by the model. The general syntax of the 14 elements of a long parameter line is described here. If used, time-varying parameter lines use only the first seven elements of a parameter line and will be referred to as a short parameter line. Three types of time-varying properties can be applied to a base parameter: blocks or trend, environmental linkage, and random deviation. Each parameter line contains:

Column	Element	Description
1	LO	Minimum value for the parameter
2	HI	Maximum value for the parameter
3	INIT	Initial value for the parameter. If the phase (described below) for the parameter is negative the parameter is fixed at this value. If the ss3.par file is read, it overwrites these INIT values.
4	PRIOR	Expected value for the parameter. This value is ignored if the prior type is 0 (no prior) or 1 (symmetric beta). If the selected prior type (described below) is lognormal, this value is entered in log space.
5	PRIOR SD	Standard deviation for the prior, used to calculate likelihood of the current parameter value. This value is ignored if prior type is 0. The standard deviation is in regular space regardless of the prior type.
6	PRIOR TYPE	0 = none; 1 = symmetric beta; 2 = full beta; 3 = lognormal without bias adjustment; 4 = lognormal with bias adjustment; 5 = gamma; and 6 = normal.
7	PHASE	Phase in which parameter begins to be estimated. A negative value causes the parameter to retain its INIT value (or value read from the ss3.par file).
8	Env var & Link	Create a linkage to an input environmental time-series
9	Dev link	Invokes use of the deviation vector in the linkage function
10	Dev min yr	Beginning year for the deviation vector
11	Dev max yr	Ending year for the deviation vector
12	Dev phase	Phase for estimation for elements in the deviation vector
13	Block	Time block or trend to be applied
14	Block function	Functional form for the block offset.

Note that relative to Stock Synthesis v.3.24, the order of PRIOR SD and PRIOR TYPE have been switched and the PRIOR TYPE options have been renumbered.

The full parameter line (14 in length) syntax for the mortality-growth, spawn-recruitment, catchability, and selectivity sections provides additional controls to give the parameter time-varying properties. If a parameter (a full parameter line of length 14) is set up to be time-varying (i.e., parameter time blocks, annual deviations), short parameter lines, the first 7 elements, are required to be specified immediately after the main parameter block (i.e., mortality-growth parameter section). Additional information regard time-varying parameters and how to implement them is in the Using Time-Varying Parameters section.

Terminology

The term COND appears in the "Typical Value" column of this documentation (it does not actually appear in the model files), it indicates that the following section is omitted except under certain conditions, or that the factors included in the following section depend upon certain conditions. In most cases, the description in the definition column is the same as the label output to the ss_new files.

Beginning of Control File Inputs

Typical Value	Description and Options
#C comment	Comments beginning with #C at the top of the file will be retained and included in output.
0	0 = Do not read the weight-at-age (wtatage.ss) file; 1 = Read the weight-at-age (wtatage.ss) file, also read and use the growth parameters; and 2 = Future option to read the weight-at-age (wtatage.ss) file, then omit reading and using growth parameters and all length-based data. Additional information on the weight-at-age file and the expected formatting can be found in the Empirical Weight-at-Age section.

Typical Value	Description and Options
1	Number (N) of growth patterns (GP), also referred to as morphs: These are collections of fish with unique biological characteristics (growth, mortality, weight-length, reproduction). The GP x Sex x Settlement Events constitute unique growth patterns that are tracked in SS3. They are assigned these characteristics at birth and retain them throughout their lifetime. At recruitment, growth pattern members are distributed across areas (if any) and they retain their biological characteristics even if they move to another area in which a different cohort with different biological characteristics might predominate. For example, one could assign a fast-growing growth pattern to recruit predominately in a southern areas and a slow-growing growth pattern to a northern area. The natural mortality and growth parameters are specified for each growth pattern in the mortality-growth parameter section in the order of females growth pattern 1 to growth pattern N followed by males growth pattern 1 to growth pattern N in a two sex model.
1	Number of platoons within a growth pattern/morph: This allows exploration of size-dependent survivorship. A value of 1 will not create additional platoons. Odd-numbered values (i.e., 3, 5) will break the overall morph into that number of platoons creating a smaller, larger, and mean growth platoon. The higher the number of platoons the slower the model will run, so values above 5 not advised. The fraction of each morph assigned to each platoon is custom-input or designated to be a normal approximation. When multiple platoons are designated, an additional input is the ratio of between platoon to within platoon variability in size-at-age. This is used to partition the total growth variability. For the platoons, their size-at-age is calculated as a factor (determined from the between-within variability calculation) times the size-at-age of the central morph which is determined from the growth parameters for that Growth Pattern x Sex.
COND > 1	Following 2 lines are conditional on N platoons > 1.
0.7	Platoon within/between standard deviation ratio. Ratio of the amount of variability in length-at-age within platoons to between platoons so that a small ratio means that the platoons are narrower and more widely spaced. A parameter (after movement parameters) is needed if the within/between standard deviation ratio is negative.

Typical Value	Description and Options
0.2 0.6 0.2	Distribution among platoons. Enter either a custom vector or enter a vector of length N with the first value of -1 to get a normal approximation: (0.15, 0.70, 0.15) for 3 platoons, or 5 platoons (0.031, 0.237, 0.464, 0.237, 0.031).

Weight-at-Age

The capability to read empirical body weight-at-age for the population and each fleet was added starting in v.3.04, in lieu of generating these weights internally from the growth parameters, weight-at-length, and size-selectivity. The values are read from a separate file named, wtatage.ss. This file is only required to exist if this option is selected. See the Empirical Weight-at-Age section for additional information on file formatting for empirical weight-at-age.

Settlement Timing for Recruits and Distribution

In older versions of SS3 one value of spawning biomass was calculated annually at the beginning of one specified spawning season and this spawning biomass produced one annual total recruitment value. The annual recruitment value was then distributed among seasons, areas, and growth types according to other model parameters.

Additional control of the seasonal timing was added in v.3.30 and now there now is an explicit elapsed time between spawning and recruitment. Spawning still occurs, just once per year, which defines a single spawning biomass for the stock-recruitment curve but its timing can be at any specified time, not just the beginning of a season. Recruitment of the progeny from an annual spawning can now enter the population in one or more settlement events, at some point after spawning as defined by the user.

Typical Value	Description and Options
1	Recruitment distribution method. This section controls which combinations of growth pattern x area x settlement will get a portion of the total recruitment coming from each spawning. Options: 1 = no longer available (used the Stock Synthesis v.3.24 or earlier setup); 2 = main effects for growth pattern, settle timing, and area; 3 = each settle entity; and

Typical Value	Description and Options
	4 = none, no parameters (only if growth pattern x settlement x area = 1).
1	Spawner-Recruitment (not implement yet, but required), options: 1 = global; and
	2 = by area (by area is not yet implemented; there is a conceptual challenge to doing the equilibrium calculation when there is fishing).
1	Number of recruitment settlement assignments. Must be at least 1 even if only 1 settlement and 1 area because the timing of that settlement must be specified.
0	Future feature, not implement yet but required.

Growth Pattern	Month	Area	Age at settlement
1	5.5	1	0

The above example specifies settlement to mid-May (month 5.5). Note that normally the calendar age at settlement is 0 if settlement happens between the time of spawning and the end of that year, and at age 1 if settlement is in the year after spawning.

Below is an example set-up where there are multiple settlement events, with one occurring the following year after spawning:

3	Number of recruitment settlement events
0	Unused option

Growth Pattern	Month	Area	Age (for each settlement assignment)
1	11.0	1	0
1	12.0	1	0
1	1.0	1	1

Details regarding settlement of recruits and timing:

- Recruitment happens in specified settlement events (growth pattern, month, area).

- Number of unique settlement timings is calculated at runtime.

- Now there is explicit elapsed time between spawning and recruitment.

- Growth and natural mortality of the platoon begins at time of settlement, which is its real age 0.0 for growth; but pre-settlement fish exist from the beginning of the season of settlement, so can be caught if selected.

- Age at recruitment now user-controlled (should be 0 if in year of spawning).

- All fish become integer age 1 (for age determination) on their first January 1st.

- Recruitment can occur >12 months after spawning which is achieved by setting the settlement age to a value greater than 1.0.

The distribution of recruitment among these settlement events is controlled by recruitment apportionment parameters. There must be a parameter line for each growth pattern, then for each area, then for each settlement. All of these are required, but only those growth pattern x area x settlements designated to receive recruits in the recruitment design matrix will have the parameter used in the recruitment distribution calculation. For the recruitment apportionment, the parameter values are the natural log of apportionment weight. The sum of all apportionment weights is calculated for each growth pattern x area x settlements that have been designated to receive recruits in the recruitment design matrix. Then the apportionment weights are scaled to sum to 1.0 so that the total recruitment from the spawning event is distributed among the cells designated to receive recruitment. Additionally, these distribution parameters can be time-varying, so the fraction of the recruits that occur in a particular growth pattern, area, or settlement can change from year to year. To specify annual variation in the distribution or recruits by area add a start and end year in the deviation min year and max year columns. Similar to the apportionment of recruits by area, one should be fixed while the other area(s) can deviate relative to the one area. If annual deviations are specified then two additional short parameter lines will be required to specify the standard error and the autocorrelation for each area with deviations.

Recruitment Distribution and Parameters

Recruits are apportioned according to:

$$\text{apportionment}_i = \frac{e^{p_i}}{\sum_{i=1}^{N} e^{p_i}} \tag{1}$$

where p_i is the proportion of recruits to area i and N is the number of settlement events. These parameters are defined in the mortality-growth parameter section.

Tips for fixing or estimating the recruitment apportionment:

- Set the value for one of these parameters, p_i, to 0.0 and not estimate it so that other parameters will be estimated (if not fixed) relative to its fixed value.

- Give the estimated parameters a min-max so they have a good range relative to the base parameter (i.e., of min = -5 and max = 5).

- In order to get a different distribution of recruitments in different years, you will need to make at least one of the recruitment distribution parameters time-varying.

In a seasonal model, all cohorts graduate to the age of 1 when they first reach January 1, even if the seasonal structure of the model has them being spawned in the late fall. In general, this means that the model operates under the assumption that all age data have been adjusted so that fish are age 0 at the time of spawning and all fish graduate to the next age on January 1. This can be problematic if the ageing structures deposit a ring at another time of year. Consequently, you may need to add or subtract a year to some of your age data to make it conform to the model expected data structure, or more ideally you may need to define the calendar year within the model to start at the beginning of the season at which ring deposition occurs. Talk with your ageing lab about their criteria for seasonal ring deposition.

Seasonal recruitment is coded to work smoothly with growth. If the recruitment occurring in each season is assigned the same growth pattern, then each seasonal cohort's growth trajectory is simply shifted along the age/time axis. At the end of the year, the early born cohorts will be larger, but all are growing with the same growth parameters, so all will converge in size as they approach their common maximum length (e.g., no seasonal effects on growth).

At the time of settlement, fish are assigned a size equal to the lower edge of the first population size bin and they grow linearly until they reach the age A1. A warning is generated if the first population length bin is greater than 10 cm as this seems an unreasonably large value for a larval fish. A1 is in terms of real age elapsed since birth. All fish advance to the next integer age on January 1, regardless of birth season. For example, consider a 2 season model with some recruitment in each season and with each season's recruits coming from the same GP. At the end of the first year, the early born fish will be larger but both of the seasonal cohorts will advance to an integer age of 1 on Jan 1 of the next year. The full growth curve is still calculated below A1, but the size-at-age used is the linear replacement. Because the linear growth trajectory can never go negative, there is no need for the additive constant to the standard deviation (necessary for the growth model used in SS2 V1.x), but the option to add a constant has been retained in the model.

Movement

Here the movement of fish between areas are defined. This is a box transfer with no explicit adjacency of areas, so fish can move from any area to any other area in each time step. While not incorporated yet, there is a desire for future versions of SS3 to have the

capability to allow sex-specific movement, and also to allow some sort of mirroring so that sexes and growth patterns can share the same movement parameters if desired.

Typical Value	Description and Options
COND: only if areas > 1	
2	Enter Number of movement definitions.
1.0	First age that moves. This value is a real number, not an integer, to allow for an in-year start to movement in a multi-season model. It is the real age at the beginning of a season, even though movement does not occur until the end of the season. For example, in a setup with two 6-month seasons a value of 0.5 will cause the age 0 fish to not move when they complete their first 6 month season of life, and then to move at the end of their second season because they start movement capability when they reach the age of 0.5 years (6 months).
1 1 1 2 4 10 1 2 2 1 4 10	Movement definitions: season, growth pattern, source area, destination, age1, and age2. The example shown here has 1 growth patterns and 2 areas with fish moving between the two areas. The rate of movement will be controlled by the movement parameters later defined in the mortality-growth parameter section. Here the age1 and age2 specify the range over which the movement parameters are interpolated with movement constant below age1 and above age2.

Two parameters will be entered later for each growth pattern, area pair, and season.

- Movement is constant at the first parameter (P1) below the specified minimum age for movement change, constant at the second parameter (P2) above maximum age for movement change, and linearly interpolated for intermediate ages.

- A movement rate parameter can be set to use the same value as the corresponding parameter for the first defined movement pattern by entering a parameter value of -9999 and a negative phase value.

- For each source area, the implicit movement parameter value is 0.0 (movement within a single area). However, this default value can be replaced if the stay movement is selected to have an explicit pair of parameters (e.g., specify movement rate for area 1 to area 1) and will require additional parameter lines.

- A constant movement rate across all ages can be accomplished by either:

- Setting both movement ages to 0, not estimating the first movement parameter, and using a second movement parameter to cover all ages from 0 to the maximum number of ages.

- Setting movement ages to any value, estimating the first movement parameter, and setting the second movement parameter to have a value of -9998 with a negative phase.

• The parameter is exponentiated so that a movement parameter value of 0 becomes 1.0.

• For each source area, all movement rates are then summed and divided by this sum so that 100% of the fish are accounted for in the movement calculations.

$$\text{rate}_i = \frac{e^{p_i}}{\sum_{j=1}^{N} e^{p_i}} \tag{2}$$

• At least one movement parameter must be fixed so that all other movement parameters are estimated relative to it. This is achieved naturally by not specifying the stay rate parameter so it has a fixed value of 0.0.

• The resultant movement rates are multiplied by season duration in a seasonal model.

Time Blocks

Typical Value	Description and Options
3	Number of block patterns. These patterns can be referred to in the parameter sections to create a separate parameter value for each block.
COND > 0:	Following inputs are omitted if the number of block patterns equals 0.
3 2 1	Blocks per pattern:
1975 1985 1986 1990 1995 2001	Beginning and ending years for blocks in design 1; years not assigned to a block period retain the baseline value for a parameter that uses this pattern.
1987 1990 1995 2001	Beginning and ending years for blocks in design 2.
1999 2002	Beginning and ending years for blocks in design 3.

Blocks and other time-vary parameter controls are operative during forecast years, so care should be taken when setting the end year of the last block in a pattern. If that end year is set to the last year in the time series, then the parameter will revert to the base value for the forecast. If the user wants to continue the last block through the forecast, it is advisable to set the last block's end year value to -2 to cause SS3 to reset it to the last year of the forecast. Using the value -1 will set the block's end year to the last year of the time series and leave the forecast at the base parameter value. Note that additional controls on time-varying parameters in forecast years are in the forecast section.

Auto-generation

Auto-generation is a useful way to automatically create the required short time-varying parameter lines which will be written in the control.ss_new file. These parameter lines can then be copied into the control file and modified as needed. As example, if you want to add a block to natural mortality, modify the block and block function entry of the mortality parameter line, ensure that auto-generation is set to 0 (for the biology section at least) and run the model without estimation. The control.ss_new file will now show the required block parameter line specification for natural mortality and this line can be copied into the main control file. Note, that if auto-generation is on (set to 0), the model will not expect to read the time-varying parameters in that section of the control file and will error out if they are present

Typical Value	Description and Options
1	Environmental/Block/Deviation adjust method for all time-varying parameters. 1 = warning relative to base parameter bounds; and 3 = no bound check. Logistic bound check form from previous SS3 versions (e.g., v.3.24) is no longer an option.
1 1 1 1 1	Auto-generation of time-varying parameter lines. Five values control auto-generation for parameter block sections: 1-biology, 2-spawn-recruitment, 3-catchability, 4-tag (future), and 5-selectivity. The accepted values are: 0 = auto-generate all time-varying parameters (no time-varying parameters are expected); 1 = read each time-varying parameter line as exists in the control file; and

Typical Value	Description and Options
	2 = read each line and auto-generate if read if the time-varying parameter value for LO = -12345. Useful to generate reasonable starting values.

Biology

Natural Mortality

Natural mortality (M) options include some options that are referenced to integer age and other options to real age since settlement. If using an option that references M to real age since settlement, M varies by age and will change by season (e.g., cohorts born early in the year will have different M than cohorts born later in the year).

Lorenzen Natural Mortality

Lorenzen natural mortality is based on the concept that natural mortality is driven by physiological and ecological processes and varies over the life cycle of a fish. So, natural mortality is scaled by the length of the fish. In this implementation, a reference age and M value are read in, and other ages will have an M scaled to its body size-at-age. However, if platoons are used, all will have the same M as their growth pattern. Lorenzen M calculation will be updated if the starting year growth parameters are active, but if growth parameters vary during the time-series, the M is not further updated. Additionally, the M is linked to the length-at-age from the growth parameters and can't be used for an empirical weight-at-age model. Be careful in using Lorenzen when there is time-varying growth.

Age-specific M Linked to Age-Specific Length and Maturity

This is an experimental option available as of v.3.30.17.

A general model for age- and sex-specific natural mortality expands a model developed by Maunder et al. (2010) and Maunder (2011) and is based on the following some assumptions:

1. M for younger fish is due mainly to processes that are functions of the size of the individuals (e.g., predation);

2. M increases after individuals become reproductively mature;

3. Maturity follows a logistic curve; and

4. M caused by senescence is either small or occurs at an age for which there are few fish alive, so it is not influential.

The model is based on combining the observation that M is inversely proportional to length for young fish (Lorenzen, 2000) and the logistic model from Lehodey et al. (2008) for older fish. Natural mortality for a given sex and age is:

$$M_{a,s} = M_{juv,s} \frac{L_{a,s}}{L_{mat*,s}}^{\lambda} + \frac{M_{mat,s} - M_{juv,s} \frac{L_{a,s}}{L_{mat*,s}}^{\lambda}}{1 + e^{\beta_s(L_{a,s} - L_{50,s})}}, \tag{3}$$

where $M_{juv,s}$ (juvenile natural mortality), λ (power), $L_{mat*,s}$ (first mature length of fish), and $M_{mat,s}$ (the mature instantaneous natural mortality rate by sex, are user inputs in long parameter lines. For suboption 1, L_{50} and and β (slope) parameters taken from the maturity relationship within the model, which must use maturity-fecundity option 1. For suboption 3, the $L_{50,s}$ (the length at which 50% of fish are mature) and β (slope) parameters are specified in long parameter lines by the user.

Note that juvenile natural mortality, $M_{juv,s}$, and first mature length of fish, $L_{mat*,s}$, inputs are by sex (and growth pattern), but it is recommended to share them across sex by using the offset option. Using offset option 2 (males offset from females) causes male parameters to be an offset to the female parameters, so a parameter value of 0.0 for a male parameter will fix the parameter as same as the female parameter. Alternatively, using offset option 1 and setting males to 0.0 and not estimating the parameter fixes the parameter at the value of the female parameter (the section on fixing male parameters the same as female parameters has more details). This fulfills an additional assumption: M caused by reproduction may differ by sex, but juvenile M is independent of sex.

The length for a given age and sex, $L_{a,s}$ is calculated within the model.

Some suggested defaults for user-provided parameter inputs are:

- $\lambda = -1.5$ from Gulland (1987)

- $M_{mat,s} = \frac{5.4}{t_{max,s}}$ from Hamel (submitted) if t_{max} is available, otherwise
  $M_{mat,s} = 4.118 K_s^{0.73} L_{inf,s}^{-0.33}$ as in Then et al. (2015)

- $M_{juv,s} = 3 W_{mat}^{-0.288}$ from Lorenzen (1996)

Age-range Lorenzen
The original implementation of Lorenzen natural mortality in Stock Synthesis uses a reference age and its associated natural mortality as inputs to determine the Lorenzen curve. However, sometimes this information is not known. The age-range Lorenzen instead uses a range of ages and the average natural mortality over them to calculate a Lorenzen natural mortality curve.

Like the original Lorenzen options, ages will have an M scaled to its body size-at-age and care should be taken when there are multiple growth patterns or time-varying growth.

Natural Mortality Options

Typical Value	Description and Options
1	Natural Mortality Options: 0 = A single parameter; 1 = N breakpoints; 2 = Lorenzen; 3 = Read age specific M and do not do seasonal interpolation; 4 = Read age specific and do seasonal interpolation, if appropriate; 5 = age-specific M linked to age-specific length and maturity (experimental); 6 = Age-range Lorenzen.
COND = 0	No additional natural mortality controls.
COND = 1	
4	Number of breakpoints. Then read a vector of ages for these breakpoints. Later, per sex x GP, read N parameters for the natural mortality at each breakpoint.
2.5 4.5 9.0 15.0	Vector of age breakpoints.
COND = 2	
4	Reference age for Lorenzen natural mortality: read one additional integer value that is the reference age. Later read one long parameter line for each sex x growth pattern that will be the M at the reference age.
COND = 3 or 4	Do not read any natural mortality parameters in the mortality growth parameter section. With option 3, these M values are held fixed for the integer age (no seasonality or birth season considerations). With option 4, there is seasonal interpolation based on real age, just as in options 1 and 2.
0.20 0.25 ... 0.20 0.23 ...	Age-specific M values where in a 2 sex model the first row is female and the second row is male. If there are multiple growth patterns female growth pattern 1-N is read first followed by males 1-N growth pattern.
COND = 5	age-specific M linked to age-specific length and maturity suboptions.

Typical Value	Description and Options
	1 = Requires 4 long parameter lines per sex x growth pattern using maturity. Must be used with maturity option 1; 2 = reserved for future option; 3 = Requires 6 long parameter lines per sex x growth pattern
COND = 6	Read two additional integer values that are the age range for average M. Later, read one long parameter line for each sex x growth pattern that will be the average M over the reference age range.
0	Minimum age of average M range for calculating Lorenzen natural mortality.
10	Maximum age of average M range for calculating Lorenzen natural mortality.

Growth

Timing

When fish recruit at the real age of 0.0 at settlement, they have body size equal to the lower edge of the first population size bin. The fish then grow linearly until they reach a real age equal to the input value "growth-at-age for L1" and have a size equal to the parameter value for L1 (the minimum length parameter). As they age further, they grow according the selected growth equation. The growth curve is calibrated to go through the size L2 parameter when they reach the age of maximum length.

Maximum Length (Linf)

If "Growth at age for L2" is set equal to 999, then the size at the L2 parameter is used as Linf.

von Bertalanffy growth function

The von Bertalanffy growth curve is parameterized as:

$$L_t = L_\infty + (L_1 - L_\infty)e^{-k(a-A_1)} \tag{4}$$

with parameters L_1, L_∞, and k. The L_∞ is calculated as:

$$L_\infty = L_1 + \frac{(L_2 - L_1)}{1 - e^{-k(A2-A1)}} \tag{5}$$

based on the input values of fixed age for first size-at-age (A_1) and fixed age for second size-at-age (A_2).

Schnute/Richards growth function

The Richards (1959) growth model as parameterized by Schnute (1981) provides a flexible growth parameterization that allows for not only asymptotic growth but also linear, quadratic or exponential growth. The Schnute/Richards growth is invoked by entering option 2 in the growth type field. The Schnute/Richards growth function uses the standard growth parameters (e.g, Lmin, Linf, and k) and a fourth parameter that is read after reading the von Bertalanffy growth coefficient parameter (k). When this fourth parameter =has a value of 1.0, it is equivalent to the standard von Bertalanffy growth curve. When this function was first introduced, it was required that A0 parameter be set to 0.0.

The Schnute/Richards growth model is parameterized as:

$$L_t = L_{MIN}^b + (L_\infty^b - L_{MIN}^b)\frac{1 - e^{-k(t-A_1)}}{1 - e^{-k(A_2-A_1)}}^{1/b} \tag{6}$$

with parameters L_{MIN}, L_{MAX}, k, and b.

The Richards model has $b < 0$, the von Bertalanffy model has $b = 1$. The general case of $b > 0$ was called the "generalized von Bertalanffy" by Schnute (1981). The Gompertz has $b = 0$, where the equation is undefined as written above and must be replaced with:

$$L_t = y_1 e\left[log(y_2/y - 1)\frac{1 - e^{-k(t-A_1)}}{1 - e{-k}(A_2 - A_1)}\right] \tag{7}$$

Thus, if b will be estimated as a free parameter, it might be necessary to include options for constraining it to different ranges.

Mean size-at-maximum age

The mean size of fish in the max age age bin depends upon how close the growth curve is to Linf by the time it reaches max age and the mortality rate of fish after they reach max age. Users specify the mortality rate to use in this calculation during the initial equilibrium year. This must be specified by the user and should be reasonably close to M plus initial F. In v.3.30, this uses the von Bertalanffy growth out to 3 times the maximum population age and decays the numbers at age by exp(-value set here). For subsequent years of the time series, the model should update the size-at-maximum age according to the weighted average mean size of fish already at maximum age and the size of fish just graduating into maximum age. Unfortunately, this updating is only happening in years with time-varying growth. This will hopefully be fixed in a the future version.

Age-specific K

This option creates age-specific K multipliers for each age of a user-specified age range, with independent multiplicative factors for each age in the range and for each growth pattern / sex. The null value is 1.0 and each age's K is set to the next earlier age's K times the value of the current age's multiplier. Each of these multipliers is entered as a full parameter line, so inherits all time-varying capabilities of full parameters. The lower end of this age range cannot extend younger than the specified age for which the first growth parameter applies. This is a beta model feature, so examine output closely to assure you are getting the size-at-age pattern you expect. Beware of using this option in a model with seasons within year because the K deviations are indexed solely by integer age according to birth year. There is no offset for birth season timing effects, nor is there any seasonal interpolation of the age-varying K.

Growth cessation

A growth cessation model was developed for the application to tropical tuna species (Maunder et al., 2018). Growth cessation allows for a linear relationship between length and age, followed by a marked reduction of growth after the onset of sexual maturity by assuming linear growth for the youngest individuals and then a logistic function to model the decreasing growth rate at older ages.

Example growth specifications:

Typical Value	Description and Options
1	Growth Model: 1 = von Bertalanffy (3 parameters); 2 = Schnute's generalized growth curve (aka Richards curve) with 3 parameters. Third parameter has null value of 1.0; 3 = von Bertalanffy with age-specific K multipliers for specified range of ages, requires additional inputs below following the placeholder for future growth feature; 4 = age-specific K. Set base K as K for age = nages and working backwards and the age-specific K = K for the next older age * multiplier, requires additional inputs below following the placeholder for future growth feature; 5 = age specific K. Set base K as K for nages and work backwards and the age-specific K = base K * multiplier, requires additional inputs below following the placeholder for future growth feature; 6 = not implemented; 7 = not implemented; and

Typical Value	Description and Options
	8 = growth cessation. Decreases the K for older fish. If implemented, the Amin and Amax parameters, the next two lines, need to be set at 0 and 999 respectively. The mortality-growth parameter section requires the base K parameter line which is interpreted as the steepness of the logistic function that models the reduction in the growth increment by age followed by a second parameter line which is the parameter related to the maximum growth rate.
1	Growth Amin (A1): Reference age for first size-at-age L1 (post-settlement) parameter. First growth parameter is size at this age; linear growth below this.
25	Growth Amax (A2): Reference age for second size-at-age L2 (post-settlement) parameter. Use 999 to treat as L infinity.
0.20	Exponential decay for growth above maximum age (plus group: fixed at 0.20 in v.3.24; should approximate initial Z). Alternative Options: -998 = Disable growth above maximum age (plus group) similar to earlier versions of SS3 (prior to v.3.24); and -999 = Replicate the simpler calculation done in v.3.24.
0	Placeholder for future growth feature.
COND = 3	Growth model: age-specific K age-specific K where the age-specific K parameter values are multipliers of the age - 1 K parameter value. For example, if the base parameter is 0.20 based on the example set-up the K parameter for age 5 is equal to 0.20 * age-5 multiplier. Subsequently, age 6 K value is equal to age 5 K (0.20 * age-5 multiplier) multiplied by the age-6 multiplier. All ages above the maximum age with age-specific K are equal to the maximum age-specific K. The age specific K values are available in the Report file in the AGE_SPECIFIC_K section.
3	Number of K multipliers to read;
5	Minimum age for age-specific K; and
6	Second age for age-specific K; and
7	Maximum age for age-specific K.

Typical Value	Description and Options
COND = 4	Growth model: age-specific K where the age-specific K parameter values are multipliers of the age + 1 K parameter value. For example, if the base parameter is 0.20 based on the example set-up the K parameter for age 7 is equal to 0.20 * age-7 multiplier. Subsequently, age 6 K value is equal to age 7 K (0.20 * age-7 multiplier) multiplied by the age-6 multiplier. All ages below the minimum age with age-specific K are equal to the minimum age-specific K. The age specific K values are available in the Report file in the AGE_SPECIFIC_K section.
3	Number of K multipliers to read;
7	Maximum age for age-specific K;
6	Second age for age-specific K; and
5	Minimum age for age-specific K.
COND = 5	Growth model: age-specific K where the age-specific K parameter values are multipliers of the base K parameter value. For example, if the base parameter is 0.20 based on the example set-up the K parameter for age 7 is equal to 0.20 * age-7 multiplier. Subsequently, age 6 K value is equal 0.20 * age-6 multiplier. The age specific K values are available in the Report file in the AGE_SPECIFIC_K section.
3	Number of K multipliers to read;
7	Maximum age for age-specific K;
6	Second age for age-specific K; and
5	Minimum age for age-specific K.
0	Standard deviation added to length-at-age: Enter 0.10 to mimic SS2 V1.xx. Recommend using a value of 0.0.
1	CV Pattern (cannot be time-varying) 0: CV=f(LAA), so the 2 parameters are in terms of CV of the distribution of length-at-age and the interpolation between these 2 parameters is a function of mean length-at-age; 1: CV=f(A), so interpolation is a function of age; 2: SD=f(LAA), so parameters define the standard deviations of length-at-age and interpolation is a function of mean length-at-age; 3: SD=f(A); and

Typical Value	Description and Options
	4: Lognormal distribution of size-at-age. Input parameters will specify the standard deviation of natural log size-at-age (e.g., entered values will typically be between 0.05 and 0.15). A bias adjustment is applied so the lognormal distribution of size-at-age will have the same mean size as when a normal distribution is used.

Maturity-Fecundity

Typical Value	Description and Options
2	Maturity Option: 1 = length logistic; 2 = age logistic; 3 = read maturity-at-age for each female growth pattern; 4 = read a fecundity x maturity-at-age vector for all ages; 5 = disabled; and 6 = read vector of length-based maturity values. Note: need to read 2 parameter lines (maturity at 50% and maturity slope) even if option 3 or 4 is selected.
COND = 3 or 4 0 0.05 0.10 ...	Maturity Option Vector of age-specific maturity or fecundity. One row of length Nages + 1 based on the maximum population age for each female growth pattern.
COND = 6 0 0.05 0.10 ...	Maturity Option Vector of length-specific maturity or fecundity, based on the population length bins. One row of length equal to the number of population length bins (defined in the data file) for each female growth pattern.
1	First Mature Age: all ages below the first mature age will have maturity set to zero. This value is overridden if maturity option is 3 or 4 or if empirical weight-at-age (wtatage.ss) is used, but still must exist here.
1	Fecundity Option (irrelevant if maturity option is 4 or wtatage.ss is used): 1 = to interpret the 2 egg parameters as linear eggs/kg on body weight (current default), so fecundity $= wt * (a + b * wt)$, so value of a=1, b=0 causes eggs to be equivalent to spawning biomass;

Typical Value	Description and Options
	2 = to set fecundity= $a * L^b$;
	3 = to set fecundity= $a * W^b$, so values of a=1, b=1 causes fecundity to be equivalent to spawning biomass;
	4 = fecundity = $a + b * L$; and
	5 = eggs = $a + b * wt$.

Hermaphroditism

Typical Value	Description and Options
0	Hermaphroditism Option:
	0 = not used;
	1 = invoke female to male age-specific function; and
	-1 = invoke male to female age-specific function.
	Note: this creates the annual, age-specific fraction that change sex, it is not the fraction that is each sex.
COND = 1 or COND = -1	Read 2 lines below if hermaphroditism is selected. Also read 3 parameters after reading the male (option 1) or female (option -1) weight-length parameter.
-1.2	Hermaphroditism Season:
	-1 to do transition at the end of each season (after mortality and before movement); and
	<positive integer> to select just one season.
	If fractional part included (optional), indicates first age that transitions (otherwise, age 1 assumed).
0.5	Include males in spawning biomass;
	0 = no males in spawning biomass;
>0	<1 = fraction of male biomass to include in spawning biomass; and
	1 = simple addition of males to females.

The hermaphroditism option requires three full parameter lines in the mortality growth section:

1. A parameter line for inflection point (in age) at when fish may begin to change sex.

2. The standard deviation.

3. The asymptote of the maximum proportion that will transition to the other sex

These parameter lines are entered directly after the weight-at-length parameters for males.

Natural Mortality and Growth Parameter Offset Method

The most common set-up for natural mortality and growth parameters for two-sex models is direct assignment (option 1) which allows for independent estimation (or fixing) of natural mortality and growth parameters by sex. Within the direct assignment option there is functionality to set male parameters equal to the corresponding female parameter if the male INIT value is set to 0 and the phase is negative.

Alternatively, there may be situations where a user wants to create direct linkages between natural mortality and growth parameters between sexes (options 2 or 3). If the parameter offset option 2 is selected, the control file still requires that all male natural mortality and growth parameters lines to be included. The natural mortality and growth parameters (e.g., k, Lmin, Lmax, CV1, CV2) for sex > 1 (typically male fish) have a value that is an exponential offset to the female natural mortality and growth parameters, e.g., $M_{\text{male}} = M_{\text{female}} * exp(M_{\text{male offset}})$. An offset parameter can be fixed at 0.0, at a non-zero value, or estimated.

Parameter offset option 3 has an offset feature for the growth CV and for natural mortality. For the growth CV, the parameter for CV at old age is an exponential offset from the parameter for CV at young age, e.g., $CV_{\text{old}} = CV_{\text{young}} * exp(CV_{\text{offset}})$. This allows for CV old to track an estimated CV young parameter. For natural mortality, if there is more than 1 natural mortality parameter, then parameters 2 and higher for the same sex and growth pattern are exponential offsets from the first natural mortality parameter. Note that it is an old feature designed to work with natural mortality option 1 (breakpoints). It may work with natural mortality options 3 and 4, but this has not been tested.

Typical Value	Description and Options
1	Parameter Offset Method:
	$1 =$ direct assignment;
	$2 =$ for each growth pattern by sex, parameter defines offset from sex 1, offsets are in exponential terms, so for example: $M_{\text{old male}} = M_{\text{old female}} * exp(M_{\text{old male}})$; and
	$3 =$ for each growth pattern by sex, parameter defines offset from growth pattern 1 sex 1. For females, given that "natM option" is breakpoint and there are two breakpoints, parameter defines offset from early age (e.g., $M_{\text{old female}} = M_{\text{young female}} * exp(M_{\text{old female}})$. For males, given that "natM option" is breakpoint and there are two breakpoints, parameter is defined as offset from females AND from early age (e.g., $M_{\text{old male}} = M_{\text{young female}} * exp(M_{\text{young male}}) * exp(M_{\text{old male}})$).

Catch Multiplier

These parameter lines are only included in the control file if the catch multiplier field in the data file is set to 1 for a fleet. The model expected catch C_{exp} by fleet is estimated by:

$$C_{exp} = \frac{C_{obs}}{c_{mult}} \tag{8}$$

where C_{obs} is the input catch by fleet (observed catch) within the data file and c_{mult} is the estimated (or fixed) catch multiplier. It has year-specific, not season-specific, time-varying capabilities. In the catch likelihood calculation, expected catch is multiplied by the catch multiplier by year and fishery to get C_{obs} before being compared to the observed retained catch as modified by the c_{mult}.

Ageing Error Parameters

These parameters are only included in the control file if one of the ageing error definitions in the data file has requested this feature (by putting a negative value for the ageing error of the age zero fish of one ageing error definition). As of v.3.30.12, these parameters now have time-varying capability. Seven additional full parameter lines are required. The parameter lines specify:

1. Age at which the estimated pattern begins (just linear below this age), this is the start age.

2. Bias at start age (as additive offset from unbiased age).

3. Bias at maximum (as additive offset from unbiased age).

4. Power function coefficient for interpolating between those 2 values (value of 0.0 produces linear interpolation in the bias).

5. Standard deviation at start age.

6. Standard deviation at max age.

7. Power function coefficient for interpolating between those 2 values.

Code for implementing vectors of mean age and standard deviation of age can be located online within the SS_miscfxn.tpl file, search for function "get_age_age" or "SS_Label_Function 45".

Sex ratio

The last line in the mortality-growth parameter section allows the user to fix or estimate the sex ratio between female and male fish at recruitment. The parameter is specified in the fraction of female fish and is applied at settlement. The default option is a sex ratio of 0.50 with this parameter not being estimated. Any composition data input as type = 3, both sexes, will be informative to the sex ratio because it scales females and males together, not separately, for this data type input. Estimation of the sex ratio is a new feature and should be done with care with the user checking that the answer is reflective of the data.

As of v.3.30.12, this parameter now has time-varying capability similar to other parameters in the mortality-growth section.

Predator Fleet Mortality

The ability to define a predator fleet was first implemented in v.3.30.18. A parameter line for predator mortality is only required if a predator fleet has been defined in the data file. For each fleet that is designated as a predator, a new parameter line is created in the mortality-growth (MGparm) section in the control file. This parameter will have the label M2_pred1, where the "1" is the index for the predator (not the index of the fleet being used as a predator). More than one predator can be included. If the model has > 1 season, it is normal to expect M2 to vary seasonally. Therefore, only if the number of seasons is greater than 1, follow each M2 parameter with number of season parameters to provide the seasonal multipliers. These are simple multipliers times M2, so at least one of these needs to have a non-estimated value. The set of multipliers can be used to set M2 to only operate in one season if desired. If there is more than one predator fleet, each will have its own seasonal multipliers. If there is only 1 season in the model, then no multiplier lines are included.

M2 is age-specific, but not sex or morph specific. The value of the M2 parameter will be distributed across ages according to the selectivity for this fleet. In this example note that "pred1" refers to the first predator in the model, note the fleet number in which that predator has been 3 configured. The resultant age-specific M2 is added to the base M to create a total age-specific M that operates in the model exactly as M has always operated.

Because M2 is a MGparm, it can be time-varying like any other MGparm. This is important because M2, as a component added to base M, will probably always need to be time-varying by blocks, random walk or linkage to external driver. A time series of M2 from an external source could be input by setting the M2 parameter to have a base value of 0.0 and linking to the time series in the environmental data section of the data file using an additive link. In addition, the relationship should have a fixed slope of 1.0 such that $M2(y) = 0.0 + 1.0 * M2\_env\_input(y)$.

Note that all existing reports of natural mortality are the total (base M + M2) natural mortality. The M2 parameter is active in the virgin year and initial equilibrium year, where the value of M2 in the start year is used. In the future, separate control of M2 for the initial equilibrium will be provided. M2 is part of the total M used in the SPR and MSY benchmark calculations. M2 is active in the forecast era, so be attentive to its configuration if it is time-varying. Testing to date shows that this M2 feature can replicate previous results using bycatch fleets.

Note that predator calculations probably will fail if tried with F Method=1 (Pope's), even though the Pope calculation is built into the hybrid F approach.

Read Biology Parameters

Next, the model reads the mortality-growth (MG) parameters in generally the following order (may vary based on selected options):

Parameter	Description
Females	Female natural mortality and growth parameters in the following order by growth pattern.
M	Natural mortality for female growth pattern 1, where the number of natural mortality parameters depends on the option selected.
COND if M option = 1	
N breakpoints	N-1 parameter lines as an exponential offsets from the previous reference age.
Lmin	Length at Amin (units in cm) for female, growth pattern 1.
Lmax	Length at Amax (units in cm) for female, growth pattern 1.
VBK	von Bertalanffy growth coefficient (units are per year) for females, growth pattern 1.
COND if growth type = 2	
Richards Coefficient	Only include this parameter if Richards growth function is used. If included, a parameter value of 1.0 will have a null effect and produce a growth curve identical to von Bertalanffy.
COND if growth type >=3	Age-Specific K
N parameter lines equal to the number K deviations for the ages specified above.	

Parameter	Description
CV young	Variability for size at age $<=$ Amin for females, growth pattern 1. Note that CV cannot vary over time, so do not set up env-link or a deviation vector. Also, units are either as CV or as standard deviation, depending on assigned value of CV pattern.
CV old	Variability for size at age $>=$ Amax for females, growth pattern 1. For intermediate ages, do a linear interpolation of CV on means size-at-age. Note that the units for CV will depend on the CV pattern and the value of mortality-growth parameter as offset. The CV value cannot vary over time.
WtLen scale	Coefficient to convert length in cm to weight in kg for females.
WtLen exp	Exponent in to convert length to weight for females.
Mat-50%	Maturity logistic inflection (in cm or years) where female maturity-at-length (or age) is a logistic function: $M_l = 1/(1 + exp(\alpha * (l_a - \beta)))$. The α is the slope, l_a is the size-at-age, and β is the inflection of the maturity curve. Value ignored for maturity option 3, 4, and 6.
Mat-slope	Logistic slope (must have negative value). Value ignored for maturity option 3, 4, and 6.
Eggs-alpha	Two fecundity parameters; usage depends on the selected fecundity option. Must be included here even if vector is read in the control section above.
Eggs-beta	
COND: growth pattern > 1	Repeat female parameters in the above order for growth pattern 2.
Males	Male natural mortality and growth parameters in the following order by growth pattern.
M	Natural mortality for male GP1, where the number of natural mortality parameters depends on the option selected.
COND if M option $= 1$	
N	N-1 parameter lines as an exponential offsets from the previous reference age.
breakpoints	

Parameter	Description
Lmin	Length at Amin (units in cm) for male, growth pattern 1. In a two sex model, fixing the INIT value a 0 will assume the same Lmin as the female parameter value.
Lmax	Length at Amax (units in cm) for male, growth pattern 1. In a two sex model, fixing the INIT value a 0 will assume the same Lmax as the female parameter value.
VBK	von Bertalanffy growth coefficient (units are per year) for males, growth pattern 1. In a two sex model, fixing the INIT value a 0 will assume the same k as the female parameter value.
COND if growth type = 2	
Richards Coefficient	Only include this parameter if Richards growth function is used. If included, a parameter value of 1.0 will have a null effect and produce a growth curve identical to Bertalanffy.
COND if growth type = 3	Age-Specific K
N parameter lines equal to the number K deviations for the ages specified above.	
CV young	Variability for size at age $<=$ Amin for males, GP1. Note that CV cannot vary over time, so do not set up env-link or a deviation vector. Also, units are either as CV or as standard deviation, depending on assigned value of CV pattern.
CV old	Variability for size at age $>=$ Amax for males, growth pattern 1. For intermediate ages, do a linear interpolation of CV on means size-at-age. Note that the units for CV will depend on the CV pattern and the value of mortality-growth parameters as offset.
WtLen scale	Coefficient to convert length in cm to weight in kg for males.
WtLen exp	Exponent to convert length to weight for males.
COND: growth pattern > 1	Repeat male parameters in the above order for growth pattern 2.
COND: Hermaphroditism	3 parameters lines define a normal distribution for the transition rate of females to males (or vice versa).
Inflect Age	Hermaphrodite inflection age.
StDev	Hermaphrodite standard deviation (in age).

Parameter	Description
Asmp Rate	Hermaphrodite asymptotic rate.
COND: Recruitment Distribution	3 parameters lines defining recruitment distribution. See Recruitment Distribution and Parameters for more details about recruitment apportionment parameterization.
Method = 2	
Recruitment Dist. GP	Recruitment apportionment by growth pattern, if multiple growth patterns, multiple entries required.
Recruitment Dist. Area	Recruitment apportionment by area, if multiple areas, multiple entries required.
Recruitment Dist. Month	Recruitment apportionment by month, if multiple months, multiple entries required.
COND: Recruitment Distribution	1 parameter line for each settlement event defining the distribution of recruitment among them. See Recruitment Distribution and Parameters for more details about recruitment apportionment parameterization.
Method = 3	
Recruitment Dist. 1	Recruitment apportionment parameter for the 1st settlement event.
Recruitment Dist. 2	Recruitment apportionment parameter for the 2nd settlement event.
Cohort growth deviation	Set equal to 1.0 and do not estimate; it is deviations from this base that matter.
2 x N selected movement pairs	Movement parameters
COND: The following lines are only required when the associated features are turned on.	
Platoon	
StDev Ratio	
Ageing	Turned on in the data file.
Error	

Parameter	Description
Catch Multiplier	For each fleet selected for this option in the data file.
Fraction female	
	Fraction female at the time of recruitment by growth pattern, if multiple growth patterns, multiple entries required.
COND: The following lines are only required when predator fleets are invoked.	
M2 Predator	Turned on in the data file.

Example format for mortality-growth parameter section with 2 sexes, 2 areas. Parameters marked with COND are conditional on selecting that feature:

LO	HI	INIT	Prior Value	<other entries>	Block Fxn	Parameter Label
0	0.50	0.15	0.1	...	0	#NatM_p_1_Fem_GP_1
0	45	21	36	...	0	#L_at_Amin_Fem_GP_1
40	90	70	70	...	0	#L_at_Amax_Fem_GP_1
0	0.25	0.15	0.10	...	0	#VonBert_K_Fem_GP_1
0.10	0.25	0.15	0.20	...	0	#CV_young_Fem_GP_1
0.10	0.25	0.15	0.20	...	0	#CV_old_Fem_GP_1
-3	3	2e-6	0	...	0	#Wtlen_1_Fem
-3	4	3	3	...	0	#Wtlen_2_Fem
50	60	55	55	...	0	#Mat50%_Fem
-3	3	-0.2	-0.2	...	0	#Mat_slope_Fem
-5	5	0	0	...	0	#Eggs/kg_inter_Fem
-50	5	0	0	...	0	#Eggs/kg_slope_wt_Fem
0	0.50	0.15	0.1	...	0	#NatM_p_1_Mal_GP_1
0	45	21	36	...	0	#L_at_Amin_Mal_GP_1
40	90	70	70	...	0	#L_at_Amax_Mal_GP_1
0	0.25	0.15	0.10	...	0	#VonBert_K_Mal_GP_1
0.10	0.25	0.15	0.20	...	0	#CV_young_Mal_GP_1
0.10	0.25	0.15	0.20	...	0	#CV_old_Mal_GP_1
-3	3	2e-6	0	...	0	#Wtlen_1_Mal
-3	4	3	3	...	0	#Wtlen_2_Mal

LO	HI	INIT	Prior Value	<other entries>	Block Type	Parameter Label
0	0	0	0	...	0	#RecrDist_GP_1
0	0	0	0	...	0	#RecrDist_Area_1
0	0	0	0	...	0	#RecrDist_Area_2
0	0	0	0	...	0	#RecrDist_Settlement_1
0.2	5	1	1	...	0	#Cohort_Grow_Dev
-5	5	-4	1	...	0	#Move_A_seas1_GP1_from_1to2 (COND)
-5	5	-4	1	...	0	#Move_B_seas1_GP1_from_1to2 (COND)
-99	99	1	0	...	0	#AgeKeyParm1 (COND)
-99	99	0.288	0	...	0	#Age_Key_Parms 2 to 5 (COND)
-99	99	0.715	0	...	0	#Age_Key_Parm6 (COND)
0.2	3.0	1.0	0	...	0	#Catch_mult_fleet1 (COND)
0.001	0.999	0.5	0.5	...	0	#Frac_Female_GP_1
-1.0	2	0	0	...	0	#PredM2_4

Setting Male Parameters Equal to Females

The model allows a short-cut for males to use the same parameter values as female fish for natural mortality, length minimum (Length at Amin), maximum length (Length at Amax), coefficient at younger ages (CV1), coefficient at older ages (CV2), and the growth coefficient (K) when using offset option = 1 (the offset section has information on the options available). If the INIT parameter value for males is set equal to 0.0 and the phase set to negative, not estimated, each of these male parameters will use the corresponding female parameter value for the males.

Time-varying Parameters

Please see the Time-Varying Parameter Specification and Setup section for details on how to set up time varying parameters. In short, additional short parameter lines will be needed after the long parameter lines. There are some additional considerations for time-varying growth.

Seasonal Biology Parameters

Seasonal effects are available for weight-length parameters, maturity, fecundity, and for the growth parameter K. The seasonal parameter values adjust the base parameter value for

that season.

$$P' = P * exp(\text{seas\_value}) \tag{9}$$

Control file continued:

Value	Description
0 0 0 0 0 0 0 0 0 0	Seasonality for selected biology parameters (not a conditional input). Read 10 integers to specify which biology parameters have seasonality: female-wtlen1, female-wtlen2, maturity1, maturity2, fecundity1, fecundity2, male-wtlen1, male-wtlen2, L1, K. Reading a positive value selects that factor for seasonality.

COND: If any factors have seasonality, then read N seasons parameters that define the seasonal offsets from the base parameter value.

<short parameter line(s)>	Read N seasons short parameter lines for each factor selected for seasonality. The parameter values define an exponential offset from the base parameter value.

Spawner-Recruitment

The spawner-recruitment section starts by specification of the functional relationship that will be used.

Control file continued:

Value	Label	Description
3	Spawner-Recruitment Relationship	The options are: 2: Ricker: 2 parameters: ln(R0) and steepness (h); 3: Standard Beverton-Holt, 2 parameters: ln(R0) and steepness; 4: Ignore steepness and no bias adjustment. Use this in conjunction with very low emphasis on recruitment deviations to get CAGEAN-like unconstrained recruitment estimates, 2 parameters, but only uses the first one; 5: Hockey stick:3 parameters: ln(R0), steepness, and $R_{\min}$) for ln(R0), fraction of virgin SSB at which inflection occurs, and the R level at SSB=0.0; 6: Beverton-Holt with flat-top beyond Bzero, 2 parameters: ln(R0) and steepness;

Value	Label	Description
		7: Survivorship function: 3 parameters: $\ln(R0)$, z_{frac}, and β, suitable for sharks and low fecundity stocks to assure recruits are $<=$ population production;
		8: Shepherd re-parameterization: 3 parameters: $\ln(R0)$, steepness, and shape parameter, c (added to v.3.30.11 and is in beta mode); and
		9: Ricker re-parameterization: 3 parameters: $\ln(R0)$, steepness, and Ricker power, γ (added to v.3.30.11 and is in beta mode).
1	Equilibrium recruitment	Use steepness in initial equilibrium recruitment calculation
		0 = none; and
		1 = use steepness (h).
0	Future Feature	Reserved for the future option to make realized σ_R a function of the stock-recruitment curve.

Equilibrium Recruitment

In principle, steepness should always be used when calculating equilibrium recruitment. This was not the default in early versions of Stock Synthesis, so has not come into common practice. The original logic, from early 1990s version of Stock Synthesis for long-lived U.S. west coast groundfish, was that fishing had not yet gone on long enough to have reduced spawning biomass enough to reduce expected recruitment noticeably for the chosen initial equilibrium year.

Steepness should be used in the equilibrium calculation whenever you believe that the initial equilibrium catch was large enough to have reduced expected recruitment below R0. Note that when using this option, the initial equilibrium catch cannot be greater than MSY. SS3 uses the identical code for initial equilibrium catch and for MSY calculation, so it is axiomatic that MSY will be $<=$ initial equilibrium catch. So, SS3 will estimate a R0 large enough to make MSY $<$ initial equilibrium catch. Alternatively, you can elect to not use this option and instead add many years to the beginning of the time series with that same level of initial catch. In this case, it is not equilibrium, so the catch can reduce the population below B_{MSY}.

Spawner-Recruitment Functions

The number of age-0 fish is related to spawning biomass according to a stock-recruitment relationship. There are a number of options for the shape of the spawner-recruitment

relationship: Beverton-Holt, Ricker, Hockey-Stick, and a survival-based stock recruitment relationship.

Beverton-Holt

The Beverton-Holt Stock Recruitment curve is calculated as:

$$R_y = \frac{4hR_0SB_y}{SB_0(1-h) + SB_y(5h-1)}e^{-0.5b_y\sigma_R^2 + \tilde{R}_y} \qquad \tilde{R}_y \sim N(0; \sigma_R^2) \tag{10}$$

where R_0 is the unfished equilibrium recruitment, SB_0 is the unfished equilibrium spawning biomass (corresponding to R_0), SB_y is the spawning biomass at the start of the spawning season during year y, h is the steepness parameter, b_y is the bias adjustment fraction applied during year y, is the standard deviation among recruitment deviations in natural log space, and is the lognormal recruitment deviation for year y. The bias-adjustment factor (Methot and Taylor, 2011) ensures unbiased estimation of mean recruitment even during data-poor eras in which the maximum likelihood estimate of the recruitment deviation is near 0.0.

Ricker

The Ricker Stock Recruitment curve is calculated as:

$$R_y = \frac{R_0SB_y}{SB_0}e^{h(1-SB_y/SB_0)}e^{-0.5b_y\sigma_R^2 + \tilde{R}_y} \qquad \tilde{R}_y \sim N(0; \sigma_R^2) \tag{11}$$

where the stock recruitment parameters have the same meaning as described above for the Beverton-Holt.

Hockey-Stick

The hockey-stick recruitment curve is calculated as:

$$R_y = join(R_{\min}R_0 + R_0\frac{SB_y}{hSB_0}(1 - R_{\min})) + R_0(1 - join)e^{-0.5b_y\sigma_R^2 + \tilde{R}_y} \qquad \tilde{R}_y \sim N(0; \sigma_R^2) \tag{12}$$

where $R_{\min}$ is the minimum recruitment level predicted at a spawning size of zero and is set by the user in the control file, h is defined as the fraction of SB_0 below which recruitment declines linearly, and $join$ is defined as:

$$join = \left[1 + e^{1000 * \frac{(SB_0 - hSB_0)}{SB_0}}\right]^{-1} \tag{13}$$

Survivorship

The survivorship stock recruitment relationship based on Taylor et al. (2013) is a stock-recruitment model that enables explicit modeling of survival between embryos and

114

age 0 recruits, and allows the description of a wide range of pre-recruit survival curves. The model is especially useful for low fecundity species that produce relatively few offspring per litter and exhibit a more direct connection between spawning output and recruitment than species generating millions of eggs.

Survival-based recruitment is constrained so that the recruitment rate cannot exceed fecundity. The relationship between survival and spawning output is based on parameters which are on a natural log scale. These are:

$$z_0 = -ln(S_0) \tag{14}$$

which is the negative of the natural log of the equilibrium survival S_0, and can be thought of as a pre-recruit instantaneous mortality rate at equilibrium, and

$$z_{\min} = -ln(S_{\max}) = z_0(1 - z_{\text{frac}}) \tag{15}$$

which is the negative of the natural log of the maximum pre-recruit survival rate ($S_{\max}$, the limit as spawning output approaches 0), and is parameterized as a function of z_{frac} (which represents the reduction in mortality as a fraction of z_0) so the expression is well defined over the parameter range $0 < z_{\text{frac}} < 1$.

Recruitment at age 0 for each year in the time series is calculated as:

$$R_y = SB_y e^{\left(-z_0 + (z_0 - z_{min})\left(1 - (SB_y/SB_0)^\beta\right)\right)} e^{\tilde{R}_y} \qquad \tilde{R}_y \sim N(0; \sigma_R^2) \tag{16}$$

where SB_y is the spawning output in year y, β is a parameter controlling the shape of density-dependent relationship between relative spawning depletion SB_y/SB_0 and pre-recruit survival (with limit $\beta < 1$), $\tilde{R}_y$ is the recruitment in year y, and σ_R is the standard deviation of recruitment in natural log space.

As implemented in Stock Synthesis, the parameters needed to apply the stock-recruitment relationship based on the pre-recruit survival are $ln(R_0)$, z_{frac}, and β. The value of $ln(R_0)$ defines the equilibrium quantities of SB_0, S_0, and z_0 for a given set of biological inputs (either estimated of fixed), regardless of the values of z_{frac} and β.

The interpretation of the quantity $z_0 = -ln(S_0)$ as pre-recruit instantaneous mortality rate at unfished equilibrium is imperfect because the recruitment in a given year is calculated as a product of the survival fraction S_y and the spawning output SB_y for that same time period so that there is not a 1-year lag between quantification of eggs or pups and recruitment at age 0, which is when recruits are calculated within Stock Synthesis. However, if age 0 or some set of youngest ages is not selected by any fishery of survey, then density dependent survival may be assumed to occur anywhere before the first appearance of any cohort in the data or model expectations. In such cases, the upper limit on survival up to age a is given by $S_{\max}e^{-aM}$.

Nevertheless, interpreting z_0 as an instantaneous mortality helps with the understanding of z_{frac}. This parameter controls the magnitude of the density-dependent increase in survival associated with a reduction in spawning output. It represents the fraction by which this mortality-like rate is reduced as spawning output is reduced from SB_0 to 0. This is approximately equal to the increase in survival as a fraction of the maximum possible increase in survival. That is:

$$z_{\text{frac}} = \frac{ln(S_{\text{max}}) - ln(S_0)}{-ln(S_0)} \approx \frac{S_{\text{max}} - S_0}{1 - S_0} \tag{17}$$

For example, if $S_0 = 0.4$, $z_{\text{frac}} = 0.8$, then the resulting fraction increase in survival is $(S_{\text{max}} - S_0)/(1 - S_0) = 0.72$.

The parameter β controls the point where survival changes fastest as a function of spawning depletion. A value of $\beta = 1$ corresponds to a linear change in natural log survival and an approximately linear relationship between survival and spawning depletion. Values of $\beta < 1$ have survival increasing fastest at low spawning output (concave decreasing survival) whereas $\beta > 1$ has the increase in survival occurring fastest closer to the unfished equilibrium (convex decreasing survival).

The steepness (h) of the spawner-recruit curve (defined as recruitment relative to R0 at a spawning depletion level of 0.2) based on pre-recruit survival can be derived from the parameters discussed above according to the relationship and associated inequality:

$$h = 0.2e^{z_0 z_{\text{frac}}(1 - 0.2^\beta)} < 0.2e^{z_0} = \frac{1}{5S_0} = \frac{SB_0}{5R_0} \tag{18}$$

Unlike the Beverton-Holt stock-recruitment relationship, recruitment can increase above R_0 for stocks that are below SB_0 and thus the steepness is not fundamentally constrained below 1. However, in many cases, steepness will be limited well below 1 by the inequality above, which implies an inverse relationship between the maximum steepness and equilibrium survival. Specifically, the inequality above bounds steepness below 1 for all cases where $S_0 > 0.2$, which are those with the lowest fecundity, an intuitively reasonable result. For example, with $S_0 = 0.4$, the steepness is limited below 0.5, regardless of the choice of z_{frac} or β. This natural limit on steepness may be one of the most valuable aspects of this stock-recruitment relationship.

Code for the survival based recruitment can be found in SS_recruit.tpl, search for "SS_Label_43.3.7 survival based".

Shepherd

The Shepherd stock recruit curve is calculated as:

$$R_y = \left(\frac{SB_y}{SB_0}\right) \frac{5h_{adj}R_0(1 - 0.2^c)}{(1 - 5h_{adj}0.2^c) + (5h_{adj} - 1)(\frac{SB_y}{SB_0})^c} e^{-0.5b_y\sigma_R^2 + \tilde{R}_y} \qquad \tilde{R}_y \sim N(0; \sigma_R^2) \tag{19}$$

116

where c is the shape parameter for the stock recruitment curve, and h_{adj} is the transformed steepness parameter defined as:

$$h_{adj} = 0.2 + \left(\frac{h - 0.2}{0.8}\right)\left(\frac{1}{5 * 0.2^c} - 0.2\right) \tag{20}$$

More details can be found in Punt and Cope (2019).

Ricker Re-parameterization

The Ricker stock recruit curve re-parameterized version. More details can be found in Punt and Cope (2019).

$$R_y = R_0 * (1 - temp) * e^{ln(5h)(1-SB_y/SB_0)^\gamma/0.8^\gamma} \tag{21}$$

where γ is the Ricker shape parameter and $temp$ is defined as:

$$temp = \begin{cases} 1 - SB_y/SB_0 & \text{if } 1 - SB_y/SB_0 > 0 \\ 0.001 & \text{if } 1 - SB_y/SB_0 \leq 0 \end{cases} \tag{22}$$

where $temp$ stabilizes recruitment at R_0 if $SB_y > SB_0$.

Spawner-Recruitment Parameter Setup

Read the required number of long parameter set-up lines (e.g., LO, HI, INIT, PRIOR, PRIOR TYPE, SD, PHASE, ..., and BLOCK TYPE). These parameters are:

Value	Label	Description
8.5	ln(R0)	Log of virgin recruitment level.
0.60	Steepness	Steepness of spawner recruitment relationship, bound by 0.2 and 1.0 for the Beverton-Holt.
COND:		If SRR = 5, 7, or 8
	3rd Parameter	Optional depending on which spawner-recruitment relationship function is used.

Value	Label	Description
0.60	σ_R	Standard deviation of natural log recruitment. This parameter has two related roles. It penalizes deviations from the spawner-recruitment curve, and it defines the offset between the arithmetic mean spawner-recruitment curve (as calculated from ln(R0) and steepness) and the expected geometric mean (which is the basis from which the deviations are calculated. Thus the value of σ_R must be selected to approximate the true average recruitment deviation. See Tuning σ_R section below for additional guidance on how to tune σ_R.
0	Regime Parameter	This replaces the R1 offset parameter. It can have a block for the initial equilibrium year, so can fully replicate the functionality of the previous R1 offset approach. The SR regime parameter is intended to have a base value of 0.0 and not be estimated. Similar to cohort-growth deviation, it serves simply as a base for adding time-varying adjustments. This concept is similar to the old environment effect on deviates feature in v.3.24 and earlier.
0	Autocorrelation	Autocorrelation in recruitment.

Example set-up of the spawner-recruitment section:

LO	HI	INIT	PRIOR	<other entries>	Block Fxn	Parameter Label
3	31	8.81	10.3	...	0	#SR_LN(R0)
0.2	1	0.61	0.70	...	0	#SR_BH_steep
0	2	0.60	0.80	...	0	#SR_sigmaR
-5	5	0	0	...	0	#SR_regime
-99	99	0	0	...	0	#SR_autocorr

Spawner-Recruitment Time-Varying Parameters

The R0, steepness, and regime shift parameters can be time-varying by blocks, trends, environmental linkages, or random deviations. Details on how to specify time-varying parameters can be found in the Time-Varying Parameter Specification and Setup section.

However, not all of these options make sense for all parameters; please see additional details in the section on Time-Varying Stock-Recruitment Considerations.

Tuning σ_R

The σ_R value is generally not estimatable and it is recommended practice to tune input σ_R values based on the variance in estimated recruitments post running SS3. The R package `r4ss` designed to read and visualize SS3 model results provides recommendations on adjusting σ_R values in the `sigma_R_info` object in the list created by the `r4ss::SS_output()` function. An alternative σ_R value is provided based on equation:

$$\sigma_R^2 = Var(\hat{r}) + \overline{SE(\hat{r}_y)}^2 \tag{23}$$

Simulation studies conducted by Methot and Taylor (2011) compared three methods of tuning σ_R with the above approach generally performed best since it accounts for variability among recruitment deviations and the uncertainty in their estimates.

Recruitment Deviation Setup

Control file continued:

Value	Label	Description
1	Do Recruitment Deviations	This selects the way in which recruitment deviations are coded: 0: None (so all recruitments come from spawner recruitment curve). 1: Deviation vector (previously the only option): the deviations during the main period are encoded as a deviation vector that enforces them to sum to zero for this period. 2: Simple deviations: the deviations do not have an explicit constraint to sum to zero, although they still should end up having close to a zero sum. The difference in model performance between options (1) and (2) has not been fully explored to date. This is the recommended option if doing MCMC (see the issue 107 in the ADMB GitHub Repository for more information on this).

Value	Label	Description
		3: Deviation vector (added in v.3.30.13) where the estimated recruitment is equal to the R0 adjusted for blocks multiplied by a simple deviation vector of unconstrained deviations. The negative log likelihood from the deviation vector is equal to the natural log of the estimated recruitment divided by the expected recruitment by year adjusted for the spawner-recruit curve, regimes, environmental parameters, and bias-adjustment. The negative log likelihood between option 2 and 3 is approximately equal. 4: Similar to option 3 but includes a penalty based on the sum of the deviations (added in v.3.30.13).
1971	Main recruitment deviations begin year	If begin year is less than the model start year, then the early deviations are used to modify the initial age composition. However, if set to be more than the population maximum age before start year, it is changed to equal to the maximum age before start year.
2017	Main recruitment deviations end year	The final year to estimate main recruitment deviations should be set to a year where information about young fish in the data becomes limited. For example, if the model end year is 2020 and the fleet/survey only starts observing fish of age 2+, the last year to estimate main recruitment deviations could be set to 2018. Years after the main period but before the end model year will be estimated as late deviations. If recruitment deviations end year is later than retro year, it is reset to equal retro year.
3	Main recruitment deviations phase.	
1	Advanced Options	0: Use default values for advanced options 1: Read values for the 11 advanced options.
COND = 1 Beginning of advanced options		
	1950	Early Recruitment Deviations Start Year: 0: skip (default); +year: absolute year (must be less than begin year of main recruitment deviations); and

Value	Label	Description
		-integer: set relative to main recruitment deviations start year.
		Note: because this is a deviation vector, it should be long enough so that recruitment deviations for individual years are not unduly constrained.
	6	Early Recruitment Deviations Phase:
		Negative value: default value to not estimate early deviations. Users may want to set to a late phase if there is not much early data.
	0	Forecast Recruitment Deviations Phase:
		0 = Default value.
		Forecast recruitment deviations always begin in the first year after the end of the main recruitment deviations. Recruitment in the forecast period is deterministic derived from the specified stock-recruitment relationship. Setting their phase to 0 causes their phase to be set to max lambda phase +1 (so that they become active after rest of parameters have converged.). However, it is possible here to set an earlier phase for their estimation, or to set a negative phase to keep the forecast recruitment deviations at a constant level.
	1	Forecast Recruitment Deviations Lambda:
		1 = Default value.
		This lambda is for the log likelihood of the forecast recruitment deviations that occur before endyr + 1. Use a larger value here if solitary, noisy data at end of time series cause unruly recruitment deviation estimation.
	1956	Last year with no bias adjustment.
	1970	First year with full bias adjustment.
	2001	Last year with full bias adjustment.
	2002	First recent year with no bias adjustment.

Value	Label	Description
		These four entries control how the bias adjustment is phased in and then phased back out when the model is searching for the maximum log likelihood.Bias adjustment is automatically turned off when in MCMC mode. For intervening years between the first and second years in this list, the amount of bias adjustment that will be applied is linearly phased in. The first year with full bias adjustment should be a few years into the data-rich period so that the model will apply the full bias-correction only to those recruitment deviations that have enough data to inform the model about the full range of recruitment variability. Defaults for the four year values: start year - 1000, start year - Nages, main recruitment deviation final year, end year + 1. See Recruitment Likelihood with Bias Adjustment for more information.
0.85		Max Bias Adjustment: > 0: value for the maximum bias adjustment during the MLE mode. -1: bias adjustment set to 1.0 for all years, including forecast, with estimated recruitment deviations (similar to MCMC). -2: bias adjustment set to 1.0 for all years from start to end model year, bias adjustment set to 0 for the forecast period -3: bias adjustment set to 0 for all model and forecast years.
0		Period For Recruitment Cycles: Use this when the model is configured to model seasons as years and there is a need to impose a periodicity to the expected recruitment level. If value is >0, then read that number of full parameter lines below define the recruitment cycle. See more information on setting up a seasons as years model in the continuous seasonal recruitment section.
-5		Minimum Recruitment Deviation: Min value for recruitment deviation. Negative phase = Default value.
5		Maximum Recruitment Deviation: Max value for recruitment deviation. Late Phase = Default value (e.g., 5)
5		Number of Explicit Recruitment Deviations to Read:

Value	Label	Description
		0: Default, do not read any recruitment deviations; Integer: read this number of recruitment deviations.

COND = If N recruitment cycle is > 0, enter N full parameter lines below.

	<parameter line>	Full parameter line for each of the N periods of recruitment cycle.

COND = If the number of recruitment deviations to read is > 0, then enter a total number of lines equal to that specified above. A year and a deviation value is expected as:

	2010 1.27	Enter year and deviation.
	2011 -0.45	Two example recruitment deviations being read. Note: the model will rescale the entire vector of recruitment deviation after reading these deviations, so by reading two positive values, all other recruitment deviations will be scaled to a small negative value to achieve a sum to zero condition before starting model estimation.
	2012 -0.25	
	2013 0.67	
	2014 0.20	
	2015 -0.11	

End of advanced options

Recruitment Eras

Conceptually, the model treats the early, data-poor period, the main data-rich period, and the recent/forecast time period as three eras along a continuum. The user has control of the break year between eras. Each era has its own vector. The early era is defined as a vector (prior to V3.10 this was a deviation vector) so it can have zeros during the earliest years not informed by data and then a few years with non-zero values without imposing a zero-centering on this collection of deviations. The main era can be a vector of simple deviations, or a deviation vector but it is normally implemented as a deviation vector so that the spawner-recruitment function is its central tendency. The last era does not force a zero-centered deviation vector so it can have zeros during the actual forecast and non-zero values in last few years of the time series. The early and last eras are optional, but their use can help prevent balancing a preponderance of negative deviations in early years against a preponderance of positive deviations in later years. When the 3 eras are used, it would be typical to turn on the main era during an early model phase, turn on the early era during a later phase, then have the last era turn on in the final phase.

Recruitment Likelihood with Bias Adjustment

For each year in the total recruitment deviation time series (early, mid, late/forecast) the contribution of that year to the log likelihood is equal to: $dev^2/(2.0*\sigma_R^2) + offset*ln(\sigma_R)$ where offset is the recruitment bias adjustment between the arithmetic and geometric mean of expected recruitment for that year. With this approach, years with a zero or small offset

value do not contribute to the second component. σ_R may be estimable when there is good data to establish the time series of recruitment deviations.

The implemented recruitment bias adjustment is based upon the work documented in Methot and Taylor (2011) and following the work of Maunder and Deriso (2003). The concept is based upon the following logic. The σ_R represents the true variability of recruitment in the population. It provides the constraining penalty for the estimates of recruitment deviations and it is not affected by data. Where data that are informative about recruitment deviations are available, the total variability in recruitment, σ_R, is partitioned into a signal (the variability among the recruitment estimates) and the residual, the variance of each recruitment estimate calculated as:

$$
SE(\hat{r}_y)^2 + SD(\hat{r})^2 = \left(\left(\frac{1}{\sigma_d^2} + \frac{1}{\sigma_R^2} \right)^{-1/2} \right)^2 + \left(\frac{\sigma_R^2}{(\sigma_R^2 + \sigma_d^2)^{1/2}} \right)^2 = \sigma_R^2 \qquad (24)
$$

Where there are no data, no signal can be estimated and the individual recruitment deviations collapse towards 0.0 and the variance of each recruitment deviation approaches σ_R. Conversely, where there highly informative data about the recruitment deviations, then the variability among the estimated recruitment deviations will approach σ_R and the variance of each recruitment deviation will approach zero. Perfect data will estimate the recruitment time series signal perfectly. Of course, we never have perfect data so we should always expect the estimated signal (variability among the recruitment deviations) to be less than the true population recruitment variability.

The correct offset (bias adjustment) to apply to the expected value for recruitment is based on the concept that a time series of estimated recruitments should be mean unbiased, not median unbiased, because the biomass of a stock depends upon the cumulative number of recruits, which is dominated by the large recruitments. The degree of offset depends upon the degree of recruitment signal that can be estimated. Where no recruitment signal can be estimated, the median recruitment is the same as the mean recruitment, so no offset is applied. Where lognormal recruitment signal can be estimated, the mean recruitment will be greater than the median recruitment. The value:

$$
b_y = \frac{E\left(SD(\hat{r}_y) \right)^2}{\sigma_R^2} = 1 - \frac{SE(\hat{r}_y)^2}{\sigma_R^2} \qquad (25)
$$

of the offset then depends upon the partitioning of σ_R into between and within recruitment variability. The most appropriate degree of bias adjustment can be approximated from the relationship among σ_R, recruitment variability (the signal), and recruitment residual error. Because the quantity and quality of data varies during a time series, the user can control

the rate at which the offset is ramped in during the early, data-poor years, and then ramped back to zero for the forecast years. On output, the mean bias adjustment during the early and main eras is calculated, comparing this value to the RMSE of estimated recruitment deviations in the report.sso file. A warning is generated if the RMSE is small and the bias adjustment is larger than 2.0 times the ratio of $RMSE^2$ to σ_R^2. Additional information on recruitment bias adjustment can be found in the Recruitment Variability and Bias Correction section.

In MCMC mode, the model still draws recruitment deviations from the lognormal distribution, so the full offset is used such that the expected mean recruitment from this lognormal distribution will stay equal to the mean from the spawner-recruitment curve. When the model reaches the MCMC and MCEVAL phases, all bias adjustment values are set to 1.0 for all active recruitment deviations because the model is now re-sampling from the full lognormal distribution of each recruitment.

Recruitment Cycle

When the model is configured such that seasons are modeled as years, the concept of season within year disappears. However, there may be reason to still want to model a repeating pattern in expected recruitment to track an actual seasonal cycle in recruitment. If the recruitment cycle factor is set to a positive integer, this value is interpreted as the number of time units in the cycle and this number of full parameter lines will be read. The cyclic effect is modeled as an exp(p) factor times R0, so a parameter value of 0.0 has nil effect. In order to maintain the same number of total recruits over the duration of the cycle, a penalty is introduced so that the cumulative effect of the cycle produces the same number of recruits as Ncycles * R0. Because the cyclic factor operates as an exponential, this penalty is different than a penalty that would cause the sum of the cyclic factors to be 0.0. This is done by adding a penalty to the parameter likelihood.

Initial Age Composition

A non-equilibrium initial age composition is achieved by setting the first year of the recruitment deviations before the model start year. These pre-start year recruitment deviations will be applied to the initial equilibrium age composition to adjust this composition before starting the time series. The model first applies the initial F level to an equilibrium age composition to get a preliminary N-at-age vector and the catch that comes from applying the F's to that vector, then it applies the recruitment deviations for the specified number of younger ages in this vector. If the number of estimated ages in the initial age composition is less than maximum age, then the older ages will retain their equilibrium levels. Because the older ages in the initial age composition will have progressively less information from which to estimate their true deviation, the start of the bias adjustment should be set accordingly.

Fishing Mortality Method

There are two approaches for estimating fishing mortality (F) in Stock Synthesis. One provides a mid-season harvest rate using Popes approximation, the other provides a season-long F rate using the Baranov catch equation. Popes harvest rate is implemented as F_Method = 1. The subsequent options model F as a season-long rate using the Baranov catch equation.

- F_Method = 2 represents F as a parameter, F_{par}, for all fishing fleets. Because it is a parameter, it's estimation is influenced by all data in the model, with retained catch and discard catch having the greatest influence depending on the standard error for these observations. Starting values for the F_{par} can be calculated in early model phases using the hybrid approach (below), then copying those F_{hyb} values into the F_{par} values beginning in a specified phase for all fleets;

- F_Method = 3 represents F as a tuned factor, F_{hyb} using Pope's and Baranov catch equations sequentially in what is termed the hybrid method. The tuning is based on the retained catch for each fishing fleet;

- F_Method = 4 is a fleet-specific superset of F_Methods 2 and 3 and is the recommended method for modeling F in Stock Synthesis.

The F_{hyb} approach works by:

a. Applying the Popes method to get a harvest rate for each fleet in the current season;

b. Converting that harvest rate to the equivalent F_{hyb}. This is exact with one fleet and approximate with multiple fleets;

c. Adjusting those F_{hyb} values over a fixed number of iterations (2-4) using the ratio of observed to calculated catch for each fleet; and

d. Proceeding with those F_{hyb} values into subsequent model steps. F_{par} and F_{hyb} values are used in the same Baranov catch equation and a catch log-likelihood is calculated based on the observed retained catch and the annual standard error of the catch.

Some notes:

- At moderate F levels, hybrid should get a near exact match to the retained catch even if the catch standard error is high. This may result in poor fit to discard catch data if it exists.

- At high F and/or with many fleets, the hybrid approach may not get to an exact match to the observed catch.

- At high F, the problem becomes quite computationally stiff for Popes approximation and the hybrid method so convergence in ADMB may slow due to more sensitive gradients in the log likelihood with regard to other model parameters. Whereas, ADMB adjusts the F_{par} values slowly over the entire sequence of iterations, so it may converge with fewer iterations.

- The hybrid approach performs best when the catch is known with high precision (standard error < 0.05) and the total F across all fleets is not much greater than natural mortality. The F as parameter approach works best when total F is high; also where a fleet has both retained catch and discarded catch, because the F is influenced by both.

- With F_Method 4, fleets with low F can apply F_{hyb} across all model phases and high F fleets can switch to F_{par} during later phases.

- It is always advisable to allow the model to start with good starting values for parameters. This can be done by specifying a later phase (>1) under the conditional input for F_Method = 2 where early phases will use the hybrid method, then switch to F as parameter in later phases and transfer the hybrid F values to the parameter initial values. Same advice for using the F_Method = 4. However, bycatch fleets do not have retained catch, so cannot use the hybrid method. They must use F_{par} with user-specified starting value starting in phase 1.

- Tests have demonstrated that the F approach has negligible impact on the variance of derived quantities, such as spawning biomass in the final year.

Control file continued:

Typical Value	Description and Options
0.2	F ballpark This value is compared to the sum of the F's for the specified year (defined on the next line). The sum is over all seasons and areas. In older versions of SS3, the lambda was automatically set to 0.0 in the final phase, the user now has control over whether to reduce the lambda in later phases.
-1990	F ballpark year Negative value disable F ballpark.
3	F Method 1 = Pope's (discrete); 2 = Baranov (continuous) F as a parameter; 3 = Hybrid F; and

Typical Value	Description and Options
	4 = Fleet-specific parameter/hybrid F (recommended).
2.9	Maximum F
	This maximum is applied within each season and area. A value of 0.9 is recommended for F Method 1, and a value of about 4 is recommended for F Methods 2 and 3.

COND: F Method = 1, no additional input for Pope's approximation.

COND: F Method = 2:	
0.10 1 0	Initial F value, phase, and the number of F detail setup lines to read (example has 0).
	For phases prior to the phase of the F value becoming active, the hybrid option will be used and the F values so calculated become the starting values for the F parameters when this phase is reached.

If F Method = 2 and N for F detail is > 0

1 1980 1 0.20 0.05 4	Fleet, year, season, F, SE, phase - these values override the catch standard error values in the data file and the overall starting F value and phase read just above.

COND: F Method = 4	
	Read list of fleets needing parameters, starting F values, and phases. To treat a fleet F as hybrid only either select a phase of 99 or do not enter a parameter line for that fleet. A parameter line is not required for all fleets and if not specified will be treated as hybrid across all phases, except for bycatch fleets which are required to have an input parameter line. Use a negative phase to set F as constant (i.e., not estimated) in v.3.30.19 and higher.

Fleet	Parameter Value	Phase
1	0.05	2
2	0.01	1 # bycatch fleet
-9999	1	1

COND: F Method = 3 or 4

Typical Value	Description and Options
4	Number of tuning iterations in hybrid fleets. A value of 3 is sufficient with a single fleet and low Fs. A value of 5 or so may be needed to match the catch near exactly when there are many fleets and high F.

Initial Fishing Mortality

Read a short parameter setup line for each fishery and season when there is non-zero eqilibirium catch in a season for the fleet (equilibrium catches are input in the catch section of the data file). The parameters are the fishing mortalities for the initial equilibrium catches. Do not try to estimate parameters for fisheries with zero initial equilibrium catch - no parameter line is needed fleets and seasons with zero equilibirium catch.

If there is catch, then give a starting value greater than zero and it generally is best to estimate the parameter in phase 1. The initial F parameter lines are ordered as shown in the example below - by season, then within a season, by fleet.

It is possible to use the initial F method to achieve an estimate of the initial equilibrium Z in cases where the initial equilibrium catch is unknown. To do this requires 2 changes to the input files:

1. Data File: Include a positive value for the initial equilibrium catch for at least one fleet, often with a higher standard error depending upon the situation.

2. Control File: Add an initial F parameter line (short parameter line) for each fleet and season with initial equilibrium catch to be estimated immediately after the Fishing Mortality set-up. It will be influenced by the early age and size comps which should have some information about the early levels of Z.

An example setup with two fishery fleets and two seasons with initial equilibrium catches:

0.1	# F ballpark
-2001	# F ballpark year (negative value to disable)
3	# F method: 1=Pope; 2=Baranov; 3=Hybrid
3	# Maximum F value
4	# Number of iterations for tuning F in hybrid method

# Initial F parameters							
LO	HI	INIT	Prior	Pr. SD	Pr. Type	Phase	Label
0	3	0.1	0	99	0	1	#InitF_seas_1_fleet_1
0	3	0.1	0	99	0	1	#InitF_seas_1_fleet_2
0	3	0.1	0	99	0	1	#InitF_seas_2_fleet_1
0	3	0.1	0	99	0	1	#InitF_seas_2_fleet_2

Catchability

Catchability is the scaling factor that relates a model quantity to the expected value for some type of data (index). Typically this is used to converted selected numbers or biomass for a fleet into the expected value for a survey or CPUE by that fleet. In SS3, the concept has been extended so that, for example, a time series of an environmental factor could be treated as a survey of the time series of deviations for some parameter. This flexibility means that a family of link functions beyond simple proportionality is needed.

For each fishery and survey with an index, enter a row with the entries as described below:

1. Fleet Number

2. Link type or index of deviation vector: An assumed functional form between Q, the expected value, and the survey observation.

   (a) 1 = simple Q, proportional assumption about Q: $y = q * x$.

   (b) 2 = mirror simple Q - this will mirror the Q value from another fleet. Mirror in Q must refer to a lower number fleet relative to the fleet with the mirrored Q (example: fleet 3 mirror fleet 2). Requires a Q parameter line for the fleet but will not be used.

   (c) 3 = Q with power, 2 parameters establish a parameter for non-linearity in survey-abundance linkage. Assumes proportional with offset and power

function: $y = qx^c$ where $q = exp(lnQ_{base}))$ thus the c is not related to expected biomass but vulnerable biomass to Q. Therefore, $c > 0$ leads to hyper-stability and $c < 0$ leads to hyper-depletion.

(d) 4 = mirror Q with offset (2 parameter lines required). The mirrored Q with offset for with be reported as base Q + offset value. Mirror in Q must refer to a lower number fleet relative to the fleet with the mirrored Q. See mirrored Q with offset below for example set up.

(e) If the parameter is for an index of a deviation vector (index units = 35), use this column to enter the index of the deviation vector to which the index is related.

3. Extra input for link information (i.e., mirror fleet)

(a) >0 = mirror the Q from another (lower numbered survey designated by referencing the fleet number)

(b) If a depletion survey, option 34, approach is being applied the following values in this column determines how phases are adjusted:

- 0 = add 1 to phases of all parameters. Only R0 active in new phase 1. Mimics the default option of previous model versions;

- 1 = only R0 active in phase 1. Then finish with no other parameters becoming active; useful for data-limited draws of other fixed parameters. Essentially, this option allows the model to mimic DB-SRA; and

- 2 = no phase adjustments, can be used when profiling on fixed R0.

4. Estimate extra standard error for an index

(a) 0 = skip (typical); and

(b) 1 = estimate a parameter that will contain an additive constant to be added to the input standard deviation of the survey variability.

5. Bias adjustment - adjusts for log-normal bias when using an informative prior on Q. NOTE: Bias adjustment to Q is ONLY applied when also using the float approach. See below for Q float specifications. An expanded bias adjustment approach is under development.

(a) 0 = no bias adjustment applied; and

(b) 1 = apply bias adjustment. Bias correction will be applied to the estimated Q value.

6. Q float

   (a) 0 = no float, parameter is estimated; and

   (b) 1 = float, analytical solution is used, but parameter line still required.

   (c) Additional information regarding the use of and appropriate application of float in Q can be found in the Float Q section below.

For a setup with a single survey, the Q setup matrix could be:

Fleet Num.	Link Type	Link Info	Extra SD	Bias Adjust	Float	Label
3	1	0	1	1	0	#Survey
-9999	0	0	0	0	0	#End Read

LO	HI	INIT	<other entries>	PHASE	<other entries>	Block Fxn	Parameter Label
-5	5	-0.12	...	1	...	0	#Survey1 LnQ base
0	0.5	0.1	...	-1	...	0	#Survey1 Extra SD

If the Q base parameter specifies that it is time-varying by the annual deviation method, short parameter lines to specify the specifications of the deviation vector come after all the base Q parameters.

Mirrored Q with offset

Below is an example setup for fleets with a mirrored Q and offset from another fleet (link type = 4):

For a setup with a single survey, the Q setup matrix could be:

Fleet Num.	Link Type	Link Info	Extra SD	Bias Adjust	Float	Label
1	1	0	1	0	0	#Fleet 1
2	4	1	0	0	0	#Fleet 2
-9999	0	0	0	0	0	#End Read

A long parameter line is expected for each link parameter (i.e. Q) and for the

LO	HI	INIT	<other entries>	PHASE	<other entries>	Block Fxn	Parameter Label
-7	5	0.51	...	1	...	0	#Fleet 1 LnQ base
0	0.5	0.1	...	-1	...	0	#Fleet 1 Extra SD
-7	5	-6	...	-1	...	0	#Fleet 2 LnQ base
-8	5	-7	...	-1	...	0	#Fleet 2 Mirror Q offset

Float Q

The use and development of float in Q has evolved over time within SS3. The original approach in earlier versions of SS3 (version 3.24 and before) is that with Q "float" the units of the survey or fishery CPUE were treated as dimensionless so the Q was adjusted within each model iteration to maintain a mean difference of 0.0 between observed and expected (usually in natural log space). In contrast, Q as a parameter (float = 0) one had the ability to interpret the absolute scaling of Q and put a prior on it to help guide the model solution. Also, with Q as a parameter the code allowed for Q to be time-varying.

Then midway through the evolution of the v.3.24 code lineage a new Q option was introduced based on user recommendations. This option allowed Q to float and to compare the resulting Q value to a prior, hence the information in that prior would pull the model solution in direction of a floated Q that came close to the prior.

Currently, in v.3.30, that float with prior capability is fully embraced. All fleets that have any survey or CPUE options need to have a catchability specification and get a base Q parameter in the list. Any of these Q's can be either:

- Fixed: by not floating and not estimating.

- Floating: not estimated as an active parameter, set phase to negative, and not capable of being time-varying. Can have a prior, or not. Future versions may allow Q to be time-varying and then rescaling that Q vector according to the float logic but this is not yet currently implemented.

- Estimated as active parameter: so not floating. Can be time-varying and can have a prior.

133

Q relates the units of the survey or CPUE to the population abundance, not the population density per unit area. But many surveys and most fishery CPUE is a proportional to mean fish density per unit area. This does not have any impact in a one area model because the role of area is absorbed into the value of Q. In a multi-area model, one may want to assert that the relative difference in CPUE between two areas is informative about the relative abundance between the areas. However, CPUE is a measure of fish density per unit area, so one may want to multiply CPUE by area before putting the data into the model so that asserting the same Q for the two areas will be informative about relative abundance.

In v.3.30.13, a new catchability option has been added that allows Q to be mirrored and to add an offset to ln(Q) of the primary area when calculating the ln(Q) for the dependent area. The offset is a parameter and, hence, can be estimated and have a prior. This option allows the CPUE data to stay in density units and the effect of relative stock area is contained in the value of the ln(Q) offset.

Catchabilty Time-Varying Parameters

Time-Varying catchability can be used. Details on how to specify time-varying parameters can be found in the Time-Varying Parameter Specification and Setup section.

Q Conversion Issues Between Stock Synthesis v.3.24 and v.3.30

In v.3.24 it was common to use the deviation approach implemented as if it was survey specific blocks to create a time-varying Q for a single survey. In some cases, only one year's deviation was made active in order to implement, in effect, a block for Q. The transition executable (sstrans.exe) cannot convert this, but an analogous approach is available in v.3.30 because true blocks can now be used, as well as environmental links and annual deviations. Also note that deviations in v.3.24 were survey specific (so no parameter for years with no survey). In v.3.30, deviations are always year-specific, so you might have a deviation created for a year with no survey.

Selectivity and Discard

For each fleet and survey, read a definition line for size selectivity and retention.

Example Setup for Selectivity:

Size Selectivity:

Pattern	Discard	Male	Special	Label
1	2	0	0	#Fishery1
1	0	0	0	#Survey1

| 0 | 0 | 0 | 0 | #Survey2 |

| Age Selectivity: | | | | |
Pattern	Discard	Male	Special	Label
11	0	0	0	#Fishery1
11	0	0	0	#Survey1
11	0	0	0	#Survey2

Pattern

Specify the size/age selectivity pattern. See the Selectivity Pattern section for user options.

Discard

Discard options:

- Option = 0: no discarding by fleet.

- Option = 1: program will read 4 retention parameters after reading the specified number of selectivity parameters and all discarded fish are assumed dead.

- Option = 2: program will read 4 retention parameters and 4 discard mortality parameters.

- Option = 3: no additional parameters are read and all fish are assumed discarded and dead.

- Option = 4: program will read 7 retention parameters for dome-shaped retention and 4 discard mortality parameters.

- Option = negative fleet number: will mirror the retention and discard mortality pattern of the lower numbered fleet.

Male

Male specific selectivity options:

- Option = 0: no male specific selectivity parameters required. Male and female selectivity will be the same.

- Option = 1: program will read 4 additional parameters to define the male selectivity relative to the female selectivity. Anytime the male selectivity is caused to be greater

than 1.0; the entire male/female matrix of selectivity values is scaled by the max so that the realized max is 1.0. Note, this may cause gradient problems.

- Option 2: main selectivity parameters define male selectivity and female selectivity is estimated as an offset from male selectivity.This alternative is preferable if female selectivity is less than male selectivity.

- Option 3: only available if the selectivity pattern is 1, 20, or 24 and it causes the male selectivity parameters to be offset from the female parameters, rather than the male selectivity being an offset from the female selectivity.

- Option 4: similar to Option 3 only with the female parameters offset from the male parameters.

Special

Special (0/value): This value is used in different ways depending on the context. If the selectivity type is to mirror another selectivity type, then put the index of that source fleet or survey here. It must refer to a lower numbered fleet/survey. If the selectivity type is 6 (linear segment), then put the number of segments here. If the selectivity type is 7, then put a 1 here to keep selectivity constant above the mean average size for old fish of morph 1. If selectivity type is 27 (cubic spline), then put the number of nodes (knots) here.

Age Selectivity

For each fleet and survey, read a definition line for age selectivity. The 4 values to be read are the same as for the size-selectivity.

As of v.3.30.15, for some selectivity patterns the user can specify the minimum age of selected fish. Most selectivity curves by default select age 0 fish (i.e., inherently specify the minimum age of selected fish as 0). However, it is fairly common for the age bins specified in the data file to start at age 1. This means that any age 0 fish selected are pooled up into the age 1' bin, which will have a detrimental effect on fitting age-composition data. In order to prevent the selection of age 0 (or older) fish, the user can specify the minimum selected age for some selectivity patterns (12, 13, 14, 16, 18, 26, or 27) in versions of v.3.30.15 and later. For example, if the minimum selected age is 1 (so that age 0 fish are not selected), selectivity pattern type can be specified as 1XX, where XX is the selectivity pattern. A more specific example is if selectivity is age-logistic and the minimum selected age desired is 1, the selectivity pattern would be specified as 112 (the regular age-logistic selectivity pattern is option 12). The user can also select higher minimum selected ages, if desired; for example, 212 would be the age-logistic selectivity pattern with a minimum selected age of 2 (so that age 0 and 1 fish are not selected).

Reading the Selectivity and Retention Parameters

Read the required number of parameter setup lines as specified by the definition lines above. The complete order of the parameter setup lines is:

1. Size selectivity for fishery 1;

2. Retention for fishery 1 (if discard specified);

3. Discard Mortality for fishery 1 (if discard specified);

4. Male offsets for size selectivity for fishery 1 (if offsets used);

5. <repeat for additional fleets and surveys>;

6. Age selectivity for fishery 1;

7. Retention for fishery 1 (if discard specified);

8. Discard Mortality for fishery 1 (if discard specified);

9. Male offsets for age selectivity for fishery 1 (if offsets used); and

10. <repeat for additional fleets and surveys>.

The list of parameters to be read from the above setup would be:

LO	HI	INIT	PRIOR	<other entries>	Block Fxn	Parameter Label
19	80	53.5	50	...	0	#SizeSel p1 fishery 1
0.01	60	18.9	15	...	0	#SizeSel p2 fishery 1
20	70	38.6	40	...	0	#Retain_L_infl_fishery 1
0.1	10	6.5	1	...	0	#Retain_L_width_fishery 1
0.001	1	0.98	1	...	0	#Retain_L_asymptote_logit_fishery 1
-10	10	1	0	...	0	#Retain_L_maleoffset_fishery 1
0.1	1	0.6	0.6	...	0	#DiscMort_L_infl_fishery 1
-2	2	0	0	...	0	#DiscMort_L_width_fishery 1
20	70	40	40	...	0	#DiscMort_L_level_old_fishery 1

LO	HI	INIT	PRIOR	\<other entries\>	Block Fxn	Parameter Label
0.1	10	1	1	...	0	#DiscMortt_L_male_offset_fishery 1
19	80	53.5	50	...	0	#SizeSel p1 survey 1
0.01	60	18.9	15	...	0	#SizeSel p2 survey 1
0	40	0	5	...	0	#AgeSel p1 fishery 1
0	40	40	5	...	0	#AgeSel p2 fishery 1
0	40	0	5	...	0	#AgeSel p1 survey 1
0	40	40	5	...	0	#AgeSel p2 survey 1
0	40	0	5	...	0	#AgeSel p1 survey 2
0	40	0	5	...	0	#AgeSel p2 survey 2

Selectivity Patterns

The currently defined selectivity patterns, and corresponding required number of parameters, are:

SIZE BASED SELECTIVITY

Pattern	N Parameters	Description
0	0	Selectivity = 1.0 for all sizes.
1	2	Logistic selectivity.
2	6	Older version of selectivity pattern 24 for backward compatibility in treatment of sex-specific scaling
5	2	Mirror another selectivity. The two parameters select bin range.
6	2 + special	Non-parametric
8	8	Double logistic, with defined peak, uses smooth joiners; special=1 causes constant selectivity above L_{inf} for morph 1. Recommend using pattern 24 instead.
9	6	Simple double logistic with no defined peak.
11	2	Selectivity = 1.0 for a specified length-bin range.
15	0	Mirror another selectivity (same as for age selectivity).
21	2	Non-parametric size selectivity

Pattern	N Parameters	Description
22	4	Double normal; similar to CASAL.
23	6	Same as the double normal pattern 24 except the final selectivity is now directly interpreted as the terminal selectivity value. Cannot be used with Pope's F method because maximum selectivity may be greater than 1.
24	6	Double normal with defined initial and final selectivity level - Recommended option.
25	3	Exponential-logistic.
27	3 + 2*N nodes	Cubic spline with N nodes.
42	5 + 2*x	Selectivity pattern 27 but with 2 additional scaling parameters.
43	4 + x	Selectivity pattern 6 but with 2 additional scaling parameters.

AGE BASED SELECTIVITY

Pattern	N Parameters	Description
0	0	Selectivity = 1.0 for ages 0+.
10	0	Selectivity = 1.0 for all ages beginning at age 1. If it is desired that age-0 fish be selected, then use pattern #11 and set minimum age to 0.0.
11	2	Selectivity = 1.0 for a specified age range.
12	2	Logistic selectivity.
13	8	Double logistic, IF joiners. Use discouraged. Use pattern 18 instead.
14	nages + 1	Separate parameter for each age (empirical), value at age is $\frac{1}{1+exp(-x)}$.
15	0	Mirror another age-specific selectivity pattern.
16	2	Coleraine single Gaussian.
17	nages + 1 or special + 1	Empirical as a random walk from previous age.

Pattern	N Parameters	Description
18	8	Double logistic, with defined peak, uses smooth joiners.
19	6	Simple double logistic with no defined peak.
20	6	Double normal with defined initial and final level. Recommended option.
26	3	Exponential logistic.
27	3 + 2*N nodes	Cubic spline in age based on N nodes.
41	2 + nages + 1	Selectivity pattern 17 but with 2 additional scaling parameters.
42	5 + 2*N nodes	Selectivity pattern 27 but with 2 additional scaling parameters.
44	4 + nages	Similar to age selectivity pattern 17 but with separate parameters for males and with revised controls.
45	4 + nages	Similar to age selectivity pattern 14 but with separate parameters for males and with revised controls.

Special Selectivity Options

Special selectivity options (type 30 in size based selectivity) are no longer specified within the control file. Specifying the use of one of these selectivity types is now done within the data file by selecting the survey "units" (see the section on Index units).

Selectivity Pattern Details

Pattern 1 (size) and 12 (age) - Simple Logistic

Logistic selectivity for the primary sex (if selectivity varies by sex) is formulated as:

$$S_l = \frac{1.0}{1 + exp(-ln(19)(L_l - p1)/p2)} \tag{26}$$

where L_l is the length bin. If age based selectivity is selected then the length bin is replaced by the age vector. If sex specific selectivity is specified the non-primary sex the p1 and p2 parameters are estimated as offsets. Note that with a large p2 parameter, selectivity may not reach 1.0 at the largest size bin. The parameters are:

- p1 - size/age at inflection; and

- p2 - width for 95% selection; a negative width causes a descending curve.

Pattern 2 (size) - Older version of selectivity pattern 24 for backward compatibility

Pattern 2 differs from pattern 24 only in the treatment of sex-specific offset parameter 5. See note in Male Selectivity Estimated as Offsets from Female Selectivity for more information. Pattern 24 was changed in v.3.30.19 with the old parameterization now provided in Pattern 2.

Pattern 5 (size) - Mirror Selectivity

Two parameters select the min and max bin number (not min max size) of the source selectivity pattern. If first parameter has value <=0, then interpreted as a value of 1 (e.g., first bin). If second parameter has value <=0, then interpreted as maximum length bin (e.g., last bin specified in the data file). The mirrored selectivity pattern must have be from a lower fleet number (e.g., already specified before the mirrored fleet).

Pattern 6 (size) - Non-parametric Selectivity

Non-parametric size selectivity uses a set of linear segments. The first way point is at Length = p1 and the last way point is at Length = p2. The total number of way points is specified by the value of the Special factor in the selectivity set-up, so the N intervals is one less than the number of way points. Intermediate way points are located at equidistant intervals between p1 and p2. Parameters 3 to N are the selectivity values at the way points, entered as logistic, e.g., $1/(1 + exp(-x))$. Ramps from 10 cm to p3 if L < p1. Constant at Np if L > p2. Note that prior to version 3.03 the way points were specified in terms of bin number, rather than length.

Pattern 8 (size) and 18 (age) - Double Logistic

Users are discouraged from using the double logistic selectivity. The double normal selectivity pattern (size pattern 24, age pattern 20) provides similar functionality but with only 6 parameters.

- p1 - peak: size (age) for peak. Should be an integer and should be at bin boundary and not estimated. But options 7 and 18 may allow estimation.

- p2 - initial: selectivity at length bin=1 (minimum length) or age=0.

- p3 - inflection: a logit transform $(1/(1 + exp(-x))$ is used so that the transformed value will be between 0 and 1. So a p1 value of -1.1 will be transformed to 0.25 and used to set the selectivity equal to 0.5 at a size (age) equal to 0.25 of the way between minimum length and peak.

- p4 - slope1: ln(slope) of left side (ascending) selectivity.

- p5 - final: logit transform for selectivity at maximum length (or maximum age).

- p6 - inflection2: logit transform for size (age) at right side selectivity equal to half way between peak + peak width and maximum length (or maximum age).

- p7 - slop2: ln(slope) of right side (descending) selectivity.

- p8 - peak width: width of flattop.

Pattern 11 (size or age) - Selectivity = 1.0 for range

Length- or age-selectivity can be set equal to 1.0 for a range of lengths or ages. If the selectivity is length-based the input parameters should match the population length bin number that will have selectivity = 1.0. A simple example how this works is as follows:

Population	Length	1	2	3	4	5	6	7	8
Bin #									
Population (cm)	Length	10	12	14	16	18	20	22	24

- p1 The first length-bin to set selectivity equal to 1.0. Using the above bin structure if p1 = 3 then selectivity = 1.0 mid-bin length of 15 cm.

- p2 The final length-bin to set selectivity equal to 1.0. Using the above bin structure if p2 = 7 then selectivity = 1.0 mid-bin length of 23 cm.

- All length bins before and after p1 and p2 will be set near zero (1e-010).

The age-selectivity approach follows that detailed above for length-selectivity but is more intuitive since parameter p1 and p2 is set in terms of population age.

Pattern 14 (age) - Revise Age

Age-selectivity pattern 14 to allow selectivity-at-age to be the same as selectivity at the next younger age. When using this option, the range on each parameter should be approximately -5 to 9 to prevent the parameters from drifting into extreme values with nil gradient. The age-based selectivity is calculated as $a = 1$ to $a = Amax + 1$:

$$S_a = \frac{1}{1 + exp(-(p_{a+1} + (9 - max(p_a))))} \tag{27}$$

Pattern 17 (age) - Random Walk

This selectivity pattern provides for a random walk in ln(selectivity). For each age $a \geq A_{min}$, where A_{min} is the minimum age for which selectivity is allowed to be non-zero,

there is a selectivity parameter, p_a, controlling the changing selectivity from age $a - 1$ to age a.

The selectivity at age a is computed as:

$$S_a = \exp(S'_a - S'_{\max}),\tag{28}$$

where

$$S'_a = \sum_{i=a_{\min}}^{A} p_i \tag{29}$$

and

$$S'_{\max} = \max\{S'_a\}.\tag{30}$$

Selectivity is fixed at $S_a = 0$ for $a < A_{\min}$.

This formulation has the properties that the maximum selectivity equals 1, positive values of p_a are associated with increasing selectivity between ages $a - 1$ and a, and negative values are associated with decreasing selectivity between those ages and $p_a = 0$ gives constant selectivity.

The condition that maximum selectivity equals 1 results in one fewer degree of freedom than the number of estimated p_a. Therefore, at least one parameter should be fixed at an arbitrary value, typically $p_{A_{\min}} = 0$.

The number of parameters lines required to the control file for pattern 17 is nages + 1, unless a greater than zero value is included in the special column. If special is greater than 0, then special + 1 is the number of parameter lines needed in the control file. The value of special should be less than or equal to nages.Input to the special column is used to reduce the number of parameters lines needed (selectivity is constant above the age represented by the last parameter value read when using special).

In typical usage:

- First parameter (for age 0) could have a value of -1000 so that the age 0 fish would get a selectivity of 0.0;

- Second parameter (for age 1) could have a value of 0.0 and not be estimated, so age 1 is the reference age against which subsequent changes occur.

- Next parameters get estimated values. To assure that selectivity increases for the younger ages, the parameter min for these parameters could be set to 0.0 or a slightly negative value.

- If dome-shaped selectivity is expected, then the parameters for older ages could have a range with the max set to 0.0 so they cannot increase further.

- To keep selectivity at a particular age the same as selectivity at the next younger age, set its parameter value to 0.0 and not estimated. This allows for all older ages to have the same selectivity.

- To keep a constant rate of change in selectivity across a range of ages, use the -999 flag to keep the same rate of change in ln(selectivity) as for the previous age.

Code for implementing random walk selectivity can be found in SS_selex.tpl, search for "SS_Label_Info_22.7.17".

Pattern 22 (size) - Double Normal with Plateau

- p1 - peak1: beginning size for the plateau (in cm).

- p2 - peak2: ending size for the plateau. Calculated as a fraction of the distance between peak1 and 99% of the lower edge of the last size bin in the model. Transformed as $(1/(1+\exp(-p2))$. So a value of 0 results in peak2 being halfway between peak1 and 99% of the last bin.

- p3 - upslope: ln(variance) on ascending side.

- p4 - downslope: ln(variance) on descending side.

Pattern 23 (size), 24 (size), 2 (legacy), and 20 (age) - Double Normal with Defined Peak and Tail Controls

- p1 - peak: beginning size (or age) for the plateau (in cm or year).

- p2 - top: width of plateau, as logistic between peak and maximum length (or age). See section on Parameter Priors regarding challenges with estimation of this parameter.

- p3 - ascending width: parameter value is ln(width).

- p4 - descending width: parameter value is ln(width).

- p5 - initial: selectivity at start bin, as logistic between 0 and 1, or a code.

- p6 - final: selectivity at last bin, as logistic between 0 and 1 (for pattern 24). For pattern 23 age selectivity at last bin, as absolute value, so can be > 1.0. Warning: do not allow this value to go above 1.0 if the F_method uses Pope's approximation. It can above 1.0 when F is in exponential form. When this parameter is above 1.0, the overall selectivity pattern will have an intermediate plateau at 1.0 (according to peak and top), then will ascend further to the final value.

Notes for Double Normal with Defined Peak and Tail Controls:

- Selectivity calculations are made based on the mid_length of a population size bin. The start bin is set to the larger of bin 1 or the first population length bin that overlaps with the data length bin.

- For the initial selectivity parameter

  - p5 > -999: use p5 with p1 and p3 to do logistic calculations,

  - p5 = -999 or -1000: do not use p5 and extend the logistic decay of small fish selectivity down to the start bin,

  - p5 < -1000: ignore the initial selectivity algorithm as above and set selectivity equal to 1.0e-06 for size bins 1 through bin = -1001 - value. So a value of -1003 would set selectivity to a nil level for bins 1 through 2 and begin using the modeled selectivity in bin 3.

- For the final selectivity parameter

  - p6 = -999 or -1000: ignore the final selectivity algorithm and simply decay the large fish selectivity according to parameter 4,

  - p6 < -1000: set selectivity constant for bins greater than bin number = -1000 - value.

- If start_bin < 1 and p5 ≥ -1000, then selectivity for lengths less than start_bin are decayed according to the square of their mid length.

Pattern 15 (size or age) - Mirror
No parameters. Whole age range is mirrored from another fleet. The mirrored selectivity pattern must have be from a lower fleet number (e.g., already specified before the mirrored fleet).

Pattern 16 (age) - Gaussian (similar to Coleraine)

- p1 - age below which selectivity declines

- p2 - scaling factor for decline

Pattern 9 (size) and 19 (age) - Simple Double Logistic

This pattern has 4 parameters and 2 fixed input values.

The shape of the selectivity is provided by the function (here in terms of age a but similarly applicable to length bin l)

$$S'_a = \begin{cases} 0 & \text{if } a < p_5, \\ \left(\frac{1}{\exp(-p_2(a-p_1))} \right) \left(1 - \frac{1}{\exp(-p_4(a-[p_6 p_1 - p_3]))} \right) & \text{if } a \geq p_5. \end{cases} \tag{31}$$

which is then rescaled by first adding a small constant to all values and then rescaling to have a maximum of 1.0:

$$S_a = (S'_a + 0.000001) / \max_{a'} \{S'_a + 0.000001\} \tag{32}$$

where

- p1 - ascending inflection age/size (in cm).

- p2 - ascending slope.

- p3 - descending inflection age/size (in cm).

- p4 - descending slope.

- p5 - age or size at first selection; this is a specification parameter, so must not be estimated. Enter integer that is age for pattern 19 and is bin number for pattern 9.

- p6 - (0/1) where a value of 0 causes the descending inflection to be a standalone parameter, and a value of 1 causes the descending inflection to be interpreted as an offset from the ascending inflection. This is a specification parameter, so must not be estimated.

Pattern 25 (size) and 26 (age) - Exponential logistic

The exponential logistic included is based on the exponential logistic selectivity detailed by Thompson (1994); however, the parameterization within SS3 differs. Explorations using this selectivity form has shown that the estimation of p1 can be highly unstable. Users are strongly encourage to use the double normal selectivity rather than the exponential logistic selectivity.

- p1 - ascending rate, min: 0.02, max: 1.0, reasonable start value: 0.1.

- p2 - peak, as fraction of way between min size and max size. Parameter min value: 0.01; max: 0.99; reasonable start value: 0.5.

- p3 - descending rate, min: 0.001, max: 0.5, reasonable start value: 0.01. A value of 0.001 provides a nearly asymptotic curve. Values above 0.2 provide strongly dome-shaped function in which the p3 and p1 parameters interact strongly.

The exponential logistic selectivity is calculated as:

$$peak = \min(L_l) + p2(\max(L_l) - \min(L_l)) \tag{33}$$

where L_l is the length bins at bin l (if age-based substitute with age bins) and:

$$S_l = \frac{e^{p3*p1(peak-L_l)}}{1 - p3(1 - e^{p1(peak-L_l)})} \tag{34}$$

Pattern 27 (size or age)- Cubic Spline

This selectivity pattern uses the ADMB implementation of the cubic spline function. This function requires input of the number of nodes, the positions of those nodes, the parameter values at those nodes, and the slope of the function at the first and last node.

An alternative to specifying or estimating the slope at the first and last nodes is to fix those values at 1e30 which will cause create a "natural.cubic spline" which allows the slope (first derivative) to be flexible but sets the curvature (second derivative) to be 0 at the first and last nodes.

The number of nodes is specified in the "special" column of the selectivity set-up. The pattern number 27 is used to invoke cubic spline for size selectivity and for age selectivity; the input syntax is identical.

For a 3 node setup, the input parameters would be:

- p1 - Code for initial set-up which controls whether or not auto-generation is applied (input options are 0, 1, 2, 10, 11, or 12) as explained below

- p2 - Gradient at the first node (should be a small positive value, or fixed at 1e30 to implement a "natural cubic spline")

- p3 - Gradient at the last node (should be zero, a small negative value, or fixed at 1e30 to implement a "natural cubic spline")

- p4-p6 - The nodes in units of cm; must be in rank order and inside of the range of the population length bins. These must be held constant (not estimated, e.g., negative phase value) during a model run.

- p7-p9 - The values at the nodes. Units are ln(selectivity) before rescaling.

Notes:

- There must be at least 3 nodes.

- One of the node selectivity parameter values should be held constant so others are estimated relative to it.

- Selectivity is forced to be constant for sizes greater than the size at the last node.

- The overall selectivity curve is scaled to have a peak equal to 1.0.

- Last node cannot be at the max population length bin.

Code for implementing cubic spline selectivity can be found in SS_selex.tpl, search for "SS_Label_Info_22.7.27".

One potential problem that may occur with a cubic spline is a U-shaped pattern in the selectivity around the first node. If this occurs, the initial set-up code (auto-generation options described below) can be changed from 0, 1 or 2 to 10, 11, or 12 which will cause selectivity to be fixed at 0.0 for all bins below the first node. A natural cubic spline (noted above) may be an alternative solution to this problem.

Auto-Generation of Cubic Spline Control File Set-Up: A new feature pioneered with the cubic spline function is a capability to produce more specific parameter labels and to auto-generate selectivity parameter setup. The auto-generation feature is controlled by the first selectivity parameter value for each fleet that is specified to use the cubic spline. There are 6 possible values for this setup parameter:

- 0: no auto-generation, process parameter setup as read.

- 1: auto-generate the node locations based on the specified number of nodes and on the cumulative size distribution of the data for this fleet/survey.

- 2: auto-generate the nodes and also the min, max, prior, initial, and phase for each parameter.

- 10: no auto-generation as in 0 above and fix selectivity = 0.0 below the first knot

- 11: auto-generate the node locations as in 1 above and also fix selectivity = 0.0 below the first knot

- 12: auto-generate the nodes and other inputs as in 2 above and also fix selectivity = 0.0 below the first knot

With either the auto-generate option 1, 2 11, or 12, it still is necessary to include in the parameter file placeholder rows of values so that the init_matrix command can input the current number of values because all selectivity parameter lines are read as a single matrix with the dimension of N parameters x 14 columns. The read values of min, max, initial, prior, prior type, prior standard deviation, and phase will be overwritten.

Cumulative size and age distribution is calculated for each fleet, summing across all samples and both sexes. These distributions are output in echoinput.sso and in a new OVERALL_COMPS section of report.sso.

When the nodes are auto-generated, the first node is placed at the size corresponding to the 2.5% percentile of the cumulative size distribution, the last is placed at the 97.5% percentile of the size distribution, and the remainder are placed at equally spaced percentiles along the cumulative size distribution. These calculated node values are output into control.ss_new. So, the user could extract these nodes from control.ss_new, edit them to desired values, then, insert them into the input control file. Remember to turn off auto-generation in the revised control file.

When the complete auto-generation is selected, the control.ss_new would look like the table below:

LO	HI	INIT	<other entries>	Block Fxn	Parameter Label
0	2	2.0	...	0	#SizeSpline Code
-0.001	1	0.13	...	0	#SizeSpline GradLo
-1	0.001	-0.03	...	0	#SizeSpline GradHi
11	95	38	...	0	#SizeSpline Knot1
11	95	59	...	0	#SizeSpline Knot2
11	95	74	...	0	#SizeSpline Knot3
-9	7	-3	...	0	#SizeSpline Value1
-9	7	-1	...	0	#SizeSpline Value2
-9	7	-0.78	...	0	#SizeSpline Value3

Pattern 41 (age) - Random Walk with User-Defined Scaling

Selectivity pattern 17 with two additional parameters. The two additional parameters are the bin numbers to define the range of bins for scaling. All of the selectivity values will be scaled (divided) by the mean value over this range. The low and high bin numbers are defined before the other selectivity parameters.

LO	HI	INIT	<other entries>	Block Fxn	Parameter Label
0	20	10	...	0	#AgeSel_ScaleAgeLo
0	20	20	...	0	#AgeSel_ScaleAgeHi

Pattern 42 (size or age) - Cubic Spline with User-Defined Scaling

Selectivity pattern 27 with two additional parameters. The two additional parameters are the bin numbers to define the range of bins for scaling. All of the selectivity values will be scaled (divided) by the mean value over this range. The low and high bin numbers are defined before the other selectivity parameters.

LO	HI	INIT	<other entries>	Block Fxn	Parameter Label
0	20	10	...	0	#AgeSpline_ScaleAgeLo
0	20	20	...	0	#AgeSpline_ScaleAgeHi

Pattern 43 (size) - Non-parametric with User-Defined Scaling

Selectivity pattern 6 with two additional parameters. The two additional parameters are the bin numbers to define the range of bins for scaling. All of the selectivity values will be scaled (divided) by the mean value over this range. The low and high bin numbers are defined before the other selectivity parameters.

LO	HI	INIT	<other entries>	Block Fxn	Parameter Label
1	80	50	...	0	#SizeSel_ScaleBinLo
1	80	70	...	0	#SizeSel_ScaleBinHi

Pattern 44 (age)

Similar to pattern 17 but with separate parameters for males and females. This selectivity pattern provides for a random walk in ln(selectivity). In typical usage:

- p1 - First parameter (for age 0) could have a value of -1000 so that the age 0 fish would get a selectivity of 0.0.

- p2 - The first age for which mean selectivity = 1.

- p3 - The last age for which mean selectivity = 1.

- p4 - Male mean selectivity relative to the female selectivity mean entered as ln(ratio) for the male relative female selectivity.

- p5 - p$_n$ - Additional parameter lines for the natural log of the selectivity change between ages corresponding to the user specified number of changes in the "special" column for the selectivity specification for each sex with females entered first then males.

- -999 input indicates to the model to keep the change unchanged from the previous age (keeps same rate of change).

- -1000 input indicates used only for male selectivity indicates to the model to set the change in male selectivity equal to the female change in selectivity.

An example specification and setup for this selectivity option where selectivity is dome-shaped, peaking at age 2 with female and male selectivity are equal with 4 change points per sex:

#Pattern	Discard	Male	Special
44	0	0	4

LO	HI	INIT	\<other entries\>	Block Fxn	Parameter Label
0	20	0	...	0	#first selex age
0	20	2	...	0	#first age peak selex (mean)
0	20	2	...	0	#last age peak selex (mean)
-1	2	-0.001	...	0	#male ln(ratio)
-10	10	3.01	...	0	#female ln(selex) change 1
-10	10	1.56	...	0	#female ln(selex) change 2
-10	10	-0.15	...	0	#female ln(selex) change 3
-10	10	-0.15	...	0	#female ln(selex) change 4
-1000	10	-1000	...	0	#male ln(selex) change 1
-1000	10	-1000	...	0	#male ln(selex) change 2
-1000	10	-1000	...	0	#male ln(selex) change 3
-1000	10	-1000	...	0	#male ln(selex) change 4

Pattern 45 (age) - Revise Age

Similar to pattern 14 but with separate parameters for males and females. Age-selectivity pattern 45 to allow selectivity-at-age to be the same as selectivity at the next younger age.

- p1 - is the first age with non-zero selectivity.

- p2 - The first age in mean for peak selectivity

- p3 - The last age in mean for peak selectivity

- p4 - The male mean selectivity relative to the female mean, entered as ln(ratio) for the male relative female selectivity

- -999 input indicates to the model to keep the change unchanged from the previous age (keeps same rate of change).

- -1000 input indicates used only for male selectivity indicates to the model to set the change in male selectivity equal to the female change in selectivity.

An example specification and setup for this selectivity option where selectivity is asymptotic, with female and male selectivity are equal with 4 change points per sex:

#Pattern	Discard	Male	Special
45	0	0	3

LO	HI	INIT	<other entries>	Block Fxn	Parameter Label
0	20	2	...	0	#first selex age
0	20	5	...	0	#first age peak selex (mean)
0	20	5	...	0	#last age peak selex (mean)
-1	2	-0.001	...	0	#male ln(ratio)
-10	10	-8.1	...	0	#female ln(selex) change 1
-10	10	-4.1	...	0	#female ln(selex) change 2
-10	10	-1.8	...	0	#female ln(selex) change 3
-1000	10	-1000	...	0	#male ln(selex) change 1
-1000	10	-1000	...	0	#male ln(selex) change 2
-1000	10	-1000	...	0	#male ln(selex) change 3

Retention

Retention is defined as a logistic function of size or age. It does not apply to surveys. Four parameters (for asymptotic retention) or seven parameters (for dome-shaped retention) are used:

- Asymptotic (4 parameters):

  - p1 - ascending inflection,

  - p2 - ascending slope,

  - p3 - maximum retention controlling the height of the asymptote (smaller values result in lower asymptotes), often a time-varying quantity to match the observed amount of discard. As of v.3.30.01, this parameter is now input in logit space ranging between -10 and 10. A fixed value of -999 would assume no retention of fish and a value of 999 would set asymptotic retention equal to 1.0,

  - p4 - male offset to ascending inflection (arithmetic, not multiplicative),

- Dome-shaped (add the following 3 parameters):

  - p5 - descending inflection,

  - p6 - descending slope, and

  - p7 - male offset to descending inflection (arithmetic, not multiplicative).

$$\text{Retention}_l = \left(\frac{P3'}{1 + e^{\frac{-(L_l - (P1 + P4*male))}{P2}}} \right) * \left(1 - \frac{1}{1 + e^{\frac{-(L_l - (P5 + P7*male))}{P6}}} \right) \tag{35}$$

where $P3' = 1/(1 + e^{-P3})$ is the asymptotic retention calculated from the $P3$ parameter which is in logit space.

Discard Mortality

Discard mortality is defined as a logistic function of size such that mortality declines from 1.0 to an asymptotic level as fish get larger. It does not apply to surveys and it does not affect the calculation of expected values for discard data. It is applied so that the total mortality rate is:

$$\text{dead fish = selectivity * (retain + (1.0-retain)*discard mortality)}$$

If discard mortality is 1.0, all selected fish are dead; if discard mortality is 0.0, only the retained fish are dead.

Four parameters are used:

- p1 - descending inflection

- p2 - descending slope

- p3 - maximum discard mortality

- p4 - male offset to descending inflection (arithmetic, not multiplicative)

Discard mortality is calculated as:

$$\text{Discard Mortality}_l = 1 - \frac{1 - P3}{1 + e^{\frac{-(L_l - (P1 + P4*male))}{P2}}} \tag{36}$$

Sex-Specific Selectivity

There are two approaches to specifying sex specific selectivity. One approach allows male selectivity to be specified as a fraction of female selectivity (or vice versa). This first approach can be used for any selectivity pattern. The other option allows for separate selectivity parameters for each sex plus an additional parameter to define the scaling of one sex's peak selectivity relative to the other sex's peak. This second approach has only been implemented for a few selectivity patterns.

Male Selectivity as a Fraction of Female Selectivity (or vice versa):

If the "domale" flag is set to 1, then the selectivity parameters define female selectivity and the offset defined below sets male selectivity relative to female selectivity. The two sexes switch roles if the "domale" flag is set to 2. Generally it is best to select the option so that the dependent sex has lower selectivity, thus obviating the need to rescale for selectivities that are greater than 1.0. Sex specific selectivity is done the same way for all size and age selectivity options.

- p1 - size (age) at which a dogleg occurs (set to an integer at a bin boundary and do not estimate).

- p2 - ln(relative selectivity) at minimum length or age = 0. Typically this will be set to a value of 0.0 (for no offset) and not estimated. It would be a rare circumstance in which the youngest/smallest fish had sex-specific selectivity.

- p3 - ln(relative selectivity) at the dogleg.

- p4 - ln(relative selectivity) at maximum length or max age.

For intermediate ages, the natural log values are linearly interpolated on size (age).

If selectivity for the dependent sex is greater than the selectivity for the first sex (which always peaks at 1.0), then the male-female selectivity matrix is rescaled to have a maximum of 1.0.

Male Selectivity Estimated as Offsets from Female Selectivity (or vice versa):

A new sex selectivity option (3 or 4) has been implemented for size selectivity patterns 1 (logistic) and 23 and 24 (double normal) or age selectivity pattern 20 (double normal age). Rather than calculate male selectivity as an offset from female selectivity, here the male selectivity is calculated by making the male parameters an offset from the female parameters (option 3), or females are offset from males with option 4. The description below applies to option 3. If the size selectivity pattern is 1 (logistic), then read 3 parameters:

- male parameter 1 is added to the first selectivity parameter (inflection)

- male parameter 2 is added to the second selectivity parameter (width of curve)

- male parameter 3 is the asymptotic selectivity

If the size selectivity pattern is 2, 20, 23 or 24 (double normal), then:

- male parameter 1 is added to the first selectivity parameter (peak)

- male parameter 2 is added to the third selectivity parameter (width of ascending side); then exp(this sum) per previous transform

- male parameter 3 is added to the fourth selectivity parameter (width of descending side); then exp(sum) per previous transform

- male parameter 4 is added to the sixth selectivity parameter (selectivity at final size bin); then 1/(1+exp(-sum)) per previous transform

- male parameter 5 in selectivity pattern 24 is the apical selectivity and the descending limb for males (see note below). For selectivity pattern 2, 20, and 23 it applies only to the apical selectivity.

Notes:

- Male selectivity offsets currently cannot be time-varying because they are offsets from female selectivity, they inherit the time-varying characteristics of the female selectivity.

- Prior to v.3.30.19 male parameter 5 in pattern 24 scaled only the apical selectivity. This sometimes resulted in strange shapes when the final selectivity, which was

shared between females and males in that parameterization, was higher than the estimated apical selectivity. For backwards compatibility to the pattern 24 parameterization prior to v.3.30.19, use selectivity pattern 2.

Dirichlet-Multinomial Error for Data Weighting

If the Dirichlet-multinomial error distribution was selected in the data file for length or age data weighting, add additional parameter line(s) immediately following the age selectivity parameter block. There should be 1 parameter line for each parameter in the data file.

For additional information about the Dirichlet-multinomial please see Thorson et al. (2017) and the detailed Data Weighting section.

The list of parameters would be something like:

LO	HI	INIT	<other entries>	Block Fxn	Parameter Label
-5	10	0.5	...	0	#ln(DM theta) Age or Length 1
-5	10	0.5	...	0	#ln(DM theta) Age or Length 2

Selectivity Time-Varying Parameters

Time-Varying selectivity can be used. Details on how to specify time-varying parameters can be found in the Time-Varying Parameter Specification and Setup section.

Two-Dimensional Auto-Regressive Selectivity (2DAR; Semi-parametric selectivity)

This features combines a baseline parametric selectivity per existing options with a matrix of auto-correlated deviations by year and age (or length) to achieve semi-parametric selectivity as described in Xu et al. (2019). For brevity, only ages will be referred to here. With 2DAR, these deviations are not in the selectivity parameters themselves. Instead, the deviations are exponentiated then used as year and age-specific multipliers on the chosen baseline selectivity.

There are a range of controls for this feature. The deviations can be applied to either the age selectivity or the length selectivity over a specified range of ages (or lengths) and years. The variance of the devs can be the same for all ages, or can be age-specific up to a specified age, then constant variance for all older ages that are within the 2DAR range. Ages outside of the 2DAR range revert to the parametric selectivity. The user can select a range of years for which devs will be created and can specify whether selectivity for years

outside this range revert to the parametric selectivity or continue with the semi-parametric from the terminal year with devs; thus devs could be defined for just one year, the first year, then mirrored for all subsequent years. Finally, the controls allow for there to be a user-specified level of auto-correlation along the age dimension separate from a user-specified level of auto-correlation in the year dimension.

When using this option for age-based selectivity, if there are not too many ages, a good choice for the baseline selectivity might be random walk selectivity (pattern 17) because it provides the most flexibility, allowing the 2DAR deviations to be used only for the annual deviations around this baseline rather than the account for misspecification of the underlying functional form. Otherwise, a simple parametric selectivity form like logistic or exponential logistic would be a reasonable choice. The 2DAR option has not yet been explored adequately to provide guidance on best practices.

Typical Value	Description and Options
1	Two-dimensional auto-regressive selectivity: 0 = Not used, 1 = Use 2DAR.
COND = 1	Then read a specification line for the first fleet that uses 2DAR, then any short parameter lines invoked by those specifications, then enter another specification for next fleet using 2DAR (if any) and its parameter lines, then finish with a specification line containing negative value for the fleet.

The specification line contains 11 values:

1. Fleet: Fleet number to which semi-parametric deviations should be added.

2. Ymin: First year with deviations.

3. Ymax: Last year with deviations.

4. Amin: First integer age (or population length bin index) with deviations.

5. Amax: Last integer age (or population length bin index) with deviations.

6. Sigma_Amax: the last age (or population length bin index) for which a separate sigma should be read. The number of sigma parameter lines to be read will be Sigma_Amax - Amin + 1. The feature allows for the expected situation in which annual variability in selectivity is higher for younger ages. For simplicity, only a single sigma parameter line needs to be read if Sigma_Amax equals Amin, or if Sigma_Amax < 0.

7. Use Rho (0/1): Use autocorrelation parameters. This invokes two parameter lines, first for year, second for age (or length). The rho parameter must be set as fixed; it is treated as a constant and cannot be estimated by SS3.

8. Len(1) / Age(2): to specify whether the deviations should be applied to length- or age-based selectivity.

9. Phase: Phase to begin estimation of the deviation parameters.

10. Before Range: How should selectivity be modeled in the years prior to Ymin? Available options are (0) apply no deviations, (1) use deviations from the first year with deviations (Ymin), and (3) use average across all years with deviations (Ymin to Ymax).

11. After Range: Similar to Before Range but defines how selectivity should be modeled after Ymax.

Following each fleet-specific specification line are short parameter lines for the standard deviation of the devs (sigma_selex) and, optionally, their autocorrelation (rho).

1. Short parameter line for sigma_selex at Amin;

2. If Sigma_Amax > Amin, then additional parameter lines for each age up to Sigma_Amax;

3. If Use Rho = 1, then one parameter line for rho in the year direction and one parameter line for rho in the age (length) direction. These must be set to have a negative phase.

If multiple fleets are specified for 2DAR, the following lines are repeated for each fleet:

Fleet	Ymin	Ymax	Amin	Amax	Sigma Amax	Use Rho	Len(1)/ Age(2)	Phase	Before Range	After Range
1	1979	2015	2	10	1	1	2	5	0	0

LO	HI	INIT	PRIOR	PRIOR SD	PRIOR TYPE	PHASE	LABEL
0	4	1	1	0.1	6	-4	#Sigma selex fleet 1, first age
0	4	1	1	0.1	6	-4	#Sigma selex fleet 1, second age
0	4	1	1	0.1	6	-4	#Sigma selex fleet 1, ... age
-1	1	0	0	0.1	6	-4	#Rho year fleet 1

-1	1	0	0	0.1	6	-4		#Rho age fleet 1	

Additional fleets (e.g., fleet 2) with 2DAR selectivity

Fleet	Ymin	Ymax	Amin	Amax	Sigma Amax	Use Rho	Len(1)/ Age(2)	Phase	Before Range	After Range
2	1979	2015	2	10	1	1	2	5	0	0

LO	HI	INIT	PRIOR	PRIOR SD	PRIOR TYPE	PHASE	LABEL
0	4	1	1	0.1	6	-4	#Sigma selex fleet 1, first age
0	4	1	1	0.1	6	-4	#Sigma selex fleet 1, second age
0	4	1	1	0.1	6	-4	#Sigma selex fleet 1, ... age
-1	1	0	0	0.1	6	-4	#Rho year fleet 1
-1	1	0	0	0.1	6	-4	#Rho age fleet 1

Terminator line of 11 in length indicates the end of parameter input lines

-9999	1	1	1	1	1	1	1	1	1	1

Tag Recapture Parameters

Specify if tagging data are being used:

Typical Value	Description and Options
1	Tagging Data Present: 0 = No tagging data, or if tagging data is present in the data file, a value of 0 here will auto-generate the tag parameter section in the control.ss_new file. 1 = Read following lines of tagging data.

COND = 1 Read the following long parameter lines:

LO	HI	INIT	PRIOR	<other entries>	Block Fxn	Parameter Label
-10	10	9	9	...	0	#TG loss init 1
-10	10	9	9	...	0	#TG loss chronic 1
1	10	2	2	...	0	#TG loss overdispersion 1
-10	10	9	9	...	0	#TG report fleet 1

LO	HI	INIT	PRIOR	\<other entries\>	Block Fxn	Parameter Label
-4	0	0	0	...	0	#TG report decay 1

If there are multiple tagging groups or multiple fleets reporting tagged fish the additional needed parameter lines should be entered in order for each parameter type (i.e, TG loss init 1, TG loss init 2, TG loss chronic 1, TG loss chronic 2,..., TG report decay 1, TG report decay 2).

Five parameter types are required for tagging data. Both the tag loss parameters and the reporting rate parameters are input as any real number and a logistic transformation is used to keep the resulting rates between 0 and 1:

- Initial tag loss, representing fraction of tags that are shed or associated with tag-induced mortality immediately after tagging (1 parameter per tag group).

- Chronic tag loss, representing annual rate of tag loss (1 parameter per tag group).

- Overdispersion parameter for the negative binomial likelihood associated with the total number of tag recaptures per tag group (1 parameter per tag group). This value represents the ratio of the variance to the mean of the distribution. As parameterized in ADMB, the likelihood is only defined for overdispersion parameters > 1, but a value of 1.001 will produce results essentially equal to those associated with a Poisson likelihood in which the variance is equal to the mean. This parameter is not transformed like the other 4 types.

- Initial reporting rate (1 parameter per fleet), the reporting rate of tags for a specific fleet.

- Reporting rate decay representing annual reduction in reporting rate (1 parameter per fleet).

The tagging reporting rate parameter is transformed during estimation to maintain a positive value and is reported according to the transformation:

$$\text{Tagging Reporting Rate} = \frac{e^{\text{input parameter}}}{1 + e^{\text{input parameter}}} \tag{37}$$

Currently, tag parameters cannot be time-varying.

A shortcoming was identified in the recapture calculations when using Pope's F Method and multiple seasons in SS3 prior to v.3.30.14. The internal calculations were corrected in

v.3.30.14. Now the Z-at-age is applied internally for calculations of fishing pressure on the population when using the Pope calculations.

Mirroring of Tagging Parameters

In v.3.30.14, the ability to mirror the tagging parameters from another tag group or fleet was added. With this approach, the user can have just one parameter value for each of the five tagging parameter types and mirror all other parameters. Note that parameter lines are still required for the mirrored parameters and only lower numbered parameters can be mirrored. Mirroring is evoked through the phase input in the tagging parameter section. The options are:

- No mirroring among tag groups or fleets: phase > -1000,

- Mirror the next lower (i.e., already specified) tag group or fleet: phase = -1000 and set other parameter values the same as next lower Tag Group or fleet,

- Mirror a lower (i.e., already specified) tag group of fleet x: phase = -100x and set other parameter values the same as the mirrored tag group or fleet (i.e., if you would like to mirror fleet 1 then the phase should -1001).

To avoid having to specify mirrored parameter lines, the tag parameters can be auto-generated. The control.ss_new file created after running this model will have a full set of tagging parameter lines to use in future model runs.

Variance Adjustment Factors

When doing iterative re-weighting of the input variance factors, it is convenient to do this in the control file, rather than the data file. This section creates that capability.

Read variance adjustment factors to be applied:

Factor	Fleet	Value	Description
1	2	0.5	# Survey CV for survey/fleet 2
4	1	0.25	# Length data for fleet 1
4	2	0.75	# Length data for fleet 2
-9999	0	0	# End read

Additive Survey CV - Factor 1

While this functionality has been retained for backward compatibility with older model versions, this approach is no longer considered the best practice for tuning indices. Tuning indices should be conducted using the ability to estimate additional standard deviation which can be done in the Catchability setup.

The survey input variance (labeled survey CV) is actually the standard deviation of the ln(survey). The variance adjustment is added directly to this standard deviation. Set to 0.0 for no effect. Negative values are OK, but will crash if adjusted standard deviation value becomes negative.

Additive Discard - Factor 2

The input variance is the CV of the observation. Because this will cause observations of near zero discard to appear overly precise, the variance adjustment is added to the discard standard deviation, not to the CV. Set to 0.0 for no effect.

Additive Mean Body Weight - Factor 3

The input variance is in terms of the CV of the observation. Because such data are typically not very noisy, the variance adjustment is added to the CV and then multiplied by the observation to get the adjusted standard deviation of the observation.

Multiplicative Length Composition - Factor 4

The input variance is in terms of an effective sample size. The variance adjustment is multiplied times this sample size. Set variance adjustment to 1.0 for no effect.

Multiplicative Age Composition - Factor 5

Age composition is treated the same way as length composition.

Multiplicative Size-at-Age - Factor 6

Size-at-age input variance is the sample size for the N observations at each age. The variance adjustment is multiplied by these N values. Set to 1.0 for no effect.

Multiplicative Generalized Size Composition - Factor 7

Generalized size composition input variance is the sample size for each observation. The variance adjustment for each fleet is multiplied by these sample sizes. Set to 1.0 for no effect.

Variance Adjustment Usage Notes

The report.sso output file contains information in the "FIT_LEN_COMPS" and "FIT_AGE_COMPS" useful for determining if an adjustment of these input values is warranted to better match the scale of the average residual to the input variance scale.

Because the actual input variance factors are modified, it is these modified variance factors that are used when creating parametric bootstrap data files. So, the control files used to analyze bootstrap generated data files should have the variance adjustment factors reset to null levels.

Lambdas (Emphasis Factors)

These values are multiplied by the corresponding likelihood component to calculate the overall negative log likelihood to be minimized.

Typical Value	Description and Options
4	Max lambda phase: read this number of lambda values for each element below. The last lambda value is used for all higher numbered phases.
1	SD offset: 0 = The ln(like) to omit the $+ \ln(s)$ term, 1 = The ln(like) to include the $\ln(s)$ term for CPUE, discard, growth CV, mean body weight, recruitment deviations. If you are estimating any variance parameters, SD offset must be set to 1.

Lambda Usage Notes

If the CV for size-at-age is being estimated and the model contains mean size-at-age data, then the flag for inclusion of the + ln(stddev) term in the likelihood must be included. Otherwise, the model will always get a better fit to the mean size-at-age data by increasing the parameter for CV of size-at-age.

The reading of the lambda values has been substantially altered with v.3.30. Instead of reading a matrix containing all the needed lambda values, the model now just reads those elements that will be given a value other than 1.0. After reading the datafile, the model sets lambda equal to 0.0 if there are no data for a particular fleet/data type, and a value of 1.0 if data exist. So beware if your data files had data but you had set the lambda to 0.0 in a previous version of SS3. First read an integer for the number of changes.

You can put any placeholder value like 0 or 999 for fleet if the likelihood component is not fleet specific (like recdevs).

You can also put any placeholder value like 0 or 999 for the SizeFreq Method unless the likelihood component you are changing the lambda for is 6 = size frequency, in which case you need to have a row for each size frequency method you want to modify and put the associated method number in that fourth column.

Read the lambda adjustments by fleet and data type:

Likelihood Component	Fleet	Phase	Lambda Value	SizeFreq Method
1	2	2	1.5	1
4	2	2	10	1
10	0	2	1	0
#not_fleet_specific 2		2	1.5	1
6				
#size_frequency_method_1 2		2	1	2
6				
#size_frequency_method_2 2		3	0.2	1
4				
-9999	1	1	1	1

The codes for component are:

1 = survey	10 = recruitment deviations
2 = discard	11 = parameter priors
3 = mean weight	12 = parameter deviations
4 = length	13 = crash penalty

5 = age	14 = morph composition
6 = size frequency	15 = tag composition
7 = size-at-age	16 = tag negative binomial
8 = catch	17 = F ballpark
9 = initial equilibrium catch (see note below)	18 = regime shift

Starting in v.3.30.16, the application of a lambda to initial equilibrium catch is now fleet specific. In previous versions, a single lambda was applied in the same manner across all fleets with an initial equilibrium catch specified.

Controls for Variance of Derived Quantities

Additional standard deviation reported may be selected:

Typical Value	Description and Options
0	0 = No additional std dev reporting;
	1 = read specification for reporting stdev for selectivity, size, numbers; and
	2 = read specification for reporting stdev for selectivity, size, numbers, natural mortality, dynamic B0, and Summary Bio

COND = 1 or 2: Read the following lines (split into 3 rows for readability):

- 4 values related to selectivity:

    1. fleet number (or 0 if not used),

    2. type of selectivity to report: 1 = length, 2 = age, or 3 = combined (age with length),

    3. year (enter -1 for end year), and

    4. number of bins to report, list entered below (note - combined will report in age bins).

- 2 values related to growth:

    1. growth pattern (or 0 if not used), and

    2. number of ages for reporting: list entered below (note - each sex will be reported).

- 3 values related to numbers-at-age:

    1. area (enter -1 for sum of all areas), or 0 for not used,

    2. year (enter -1 for end year + 1), and

    3. number of ages to report, list entered below (note - each sex will be reported and summed across growth patterns).

COND = 2, enter the above quantities plus (available in versions 3.30.15 and higher):

- 2 additional values related to natural mortality-at-age:

   1. growth pattern (or 0 for not used), and

   2. number of ages for reporting, list entered below (note each sex will be reported).

- 1 additional value related to dynamic B0 (needed in versions 3.30.17 and higher):

   1. 0 for not used, 1 to report SSB for dynamic B0, 2 to report SSB and recruitment

- 1 additional value related to Summary Bio (needed in versions 3.30.17 and higher):

   1. 0 for not used, 1 to report

Depending upon the entered options above subsequent conditional inputs may be need.

- COND if selectivity specs number of bins to report > 0:

   – Vector of bin indexes or -1 in first bin will overwrite entered vector with auto-generated list to cover logical range

- COND if growth specs number of bins to report > 0 and empirical weight at age not used:

   – ... as for selectivity bins

- COND if numbers-at-age specs number of ages > 0:

   – ... as for selectivity bins

- COND == 2 and mortality-at-age specs number of ages > 0:

   – ... as for selectivity bins

Example Input:

2	# 0 = No additional std dev reporting;
	# 1 = read values below; and
	# 2 = read specification for reporting stdev for selectivity, size,numbers, and
	# natural mortality.

| 1 1 -1 5 | # Selectivity |
| 1 5 | # Growth |

1 -1 5	# Numbers-at-age
1 5	# M-at-age
1	# Dynamic Bzero
1	# Summary Biomass
5 15 25 35 38	# Vector with selectivity std bins (-1 in first bin to self-generate)
1 2 5 10 15	# Vector with growth std ages picks (-1 in first bin to self-generate)
1 2 5 10 15	# Vector with numbers-at-age std ages (-1 in first bin to self-generate)
1 2 5 10 15	# Vector with M-at-age std ages (-1 in first bin to self-generate)

999	#End of the control file input

Optional Inputs

Empirical Weight-at-Age (wtatage.ss)

The model has the capability to read empirical body weight at age for the population and each fleet, in lieu of generating these weights internally from the growth parameters, weight-at-length, and size-selectivity. Selection of this option is done by setting an explicit switch near the top of the control file. The values are read from a separate file named, wtatage.ss. This file is only required to exist if this option is selected.

The first value read is a single integer for the maximum age used in reading this file. So if the maximum age is 40, there will be 41 columns of weight-at-age entries to read, with the first column being for age 0. If the number of ages specified in this table is greater than maximum age in the model, the extra weight-at-age values are ignored. If the number of ages in this table is less than maximum age in the model, the weight-at-age data for the number of ages in the file is filled in for all unread ages out to maximum age.

The format of this input file is:

40				Maximum Age				

Year	Season	Sex	Growth Pattern	Birth Season	Fleet	Age-0	Age-1	...
1971	1	1	1	1	-2	0	0	0.1003
1971	1	1	1	1	-1	0.0169	0.0864	0.2495
1971	1	1	1	1	0	...	...	...
1971	1	1	1	1	1	...	...	...
1971	1	1	1	1	2	...	...	...
-9999	1	1	1	1	0	...	...	...

where:

- Fleet = -2 is age-specific fecundity (e.g., fecundity-scalar*maturity*weight-at-age), so time-varying fecundity is possible to implement.

- Fleet = -1 is population weight-at-age at middle of the season.

- Fleet = 0 is population weight-at-age at the beginning of the season.

- There must be an entry for fecundity, population weight-at-age at the middle of the season, population weight-at-age at the beginning of the season, and weight-at-age for each fleet (even if these vectors are identical in some cases).

- Fleets that do not use biomass do not need to have weight-at-age assigned.

- Growth pattern and birth season probably will never be used, but are included for completeness.

- A negative value for year will fill the table from that year through the ending year of the forecast, overwriting anything that has already been read for those years.

- Judicious use of negative years in the right order will allow user to enter blocks without having to enter a row of info for each year

- There is no internal error checking to verify that weight-at-age has been read for every fleet and for every year.

- In the future, there could be an option to use another value of the control file switch to turn off all aspects of growth parameters and size selectivity.

- The values entered for endyr + 1 will be used for the benchmark calculations and for the forecast; this aspect needs a bit more checking.

Caveats

- The growth curves is calculated from the input parameters and can still calculate size-selectivity and can still examine size-composition data.

- However, there is no calculation of weight-at-age from the growth input, so there is no way to compare the input weight-at-age from the weight-at-age derived from the growth parameters.

- If weight-at-age is read and size-selectivity is used, a warning is generated.

- If weight-at-age is read and discard/retention is invoked, then a warning is generated because of untested consequences for the body weight of discarded fish.

- Age-0 fish seem to need to have weight=0 for spawning biomass calculation (code -2).

User Testing

- Setup a model with age-maturity (option 2) and only age selectivity.

- Take the output calculation of weight-at-age and fecundity-at-age from report.sso and put into wtatage.ss (as shown above).

- Re-run the model with this input weight-at-age (Maturity_ Option 5) to see if identical results are produced relative to the run that had generated the weight-at-age from the growth parameters.

runnumber.ss

This file contains a single integer value. It is read when the program starts, incremented by 1, used when processing the profile value inputs (see below), used as an identifier in the batch output, then saved with the incremented value. Note that this incrementation may not occur if a run crashes.

profilevalues.ss

This file contains information for changing the value of selected parameters for each run in a batch. In the control file, each parameter that will be subject to modification by profilevalues.ss is designated by setting its phase to -9999.

The first value in profilevalues.ss is the number of parameters to be batched. This value MUST match the number of parameters with phase set equal to -9999 in the control file. The program performs no checks for this equality. If the value is zero in the first field, then nothing else will be read. Otherwise, the model will read runnumber * Nparameters values

and use the last Nparameters of these to replace the initial values of parameters designated with phase = –9999 in the control file.

Usage Note: If one of the batch runs crashes before saving the updated value of runnumber.ss, then the processing of the profilevalue.ss will not proceed as expected. Check the output carefully until a more robust procedure is developed. Also, this options was created before use of R became widespread. You probably can create a more flexible approach using R today.

Likelihood components

The objective function L is the weighted sum of the individual components indexed by kind of data i, and fishery/survey f as appropriate:

$$L = \sum_{i=1}^{I} \sum_{f=1}^{A_f} \lambda_{i,f} L_{i,f} + \lambda_R L_R + \sum_{\theta} \lambda_{\theta} L_{\theta} + \sum_{P} \lambda_P L_P + \lambda_{F_B} L_{F_B} + \lambda_{C_P} L_{C_P} \tag{38}$$

where L is the total objective function, i is the index of the objective function component, A_f is the number of fleets, $L_{i,f}$ is the objective function for data kind i for the fishery/survey f, $\lambda_{i,f}$ is a weighting factor for each objective function component, θ is the parameter priors, and P is the random parameter deviations.

The components of the objective function based on the model set-up and data are:

Index	Source	Data/Parameter	Error structure	
i	fishery/survey f	CPUE or Abundance index	user choice	
i	fishery f	Discard Biomass	user choice	
i	fishery/survey f	Mean body W or L (all ages)	normal	
i	fishery/survey f	Generalized size (W or L) composition	multinomial Dirichlet-multinomial	or
i	fishery/survey f	L Composition	multinomial Dirichlet-multinomial	or
i	fishery/survey f	Age Composition	multinomial Dirichlet-multinomial	or
i	fishery/survey f	Mean L (or W)-at-age	normal	
i	fishery/survey f	Tag-recapture 1	multinomial	
i	fishery/survey f	Tag-recapture 2	negative binomial	
i	fishery f	Initial equilibrium catch	lognormal	
i	fishery f	catch	lognormal	
R		Recruitment Deviations	lognormal	

P	Random parameter devs	normal
θ	Parameter priors	user choice
F_B	F ballpark penalty	
C_P	Crash Penalty	

Running Stock Synthesis

Command Line Interface

The name of the SS3 executable files often contains the phrase "safe" or "opt" (for optimized). The safe version includes checking for out of bounds values and should always be used whenever there is a change to the data file. The optimized version runs slightly faster but can result in data not being included in the model as intended if the safe version has not been run first. A file named "ss3.exe" is typically the safe version unless the result of renaming by the user.

On Mac and Linux computers, the executable does not include an extension (like .exe on Windows). Running the executable on from the DOS command line in Windows simply require typing the executable name (without the .exe extension):

```
> ss3
```

On Mac and Linux computers, the executable name must be preceded by a period and slash (unless its location has been added to the user's PATH). Note that the user may need to change permissions for Stock Synthesis to be executable before running SS3 for the first time:

```
> chmod a+x ss3
> ./ss3
```

An additional command has been added that allows users to specify the name of the .par file that is both read and output. Prior to v.3.30.22.1, the default exe name was 'ss' and the default .par file name was ss.par. The code now produces a ss3.par file by default instead of an ss.par file. The code will search for the default ss3.par file first, and then look for an ss.par file in order to have backwards compatibility and will by default output a ss3.par file (not an ss.par file). If you would like to read a differently named .par file and produce a .par file with the same name, you will need to add the "modelname" command. See the below example of using "modelname" to read and produce a .par file with the name "ss4you".:

```
> ./ss3 modelname ss4you
```

Additional ADMB commands can follow the executable name, such as "-nohess" to avoid calculating the Hessian matrix. To see a full list of options, add "-?" after the executable name (with a space in between).

On all operating systems, a copy of the SS3 executable can either be located in the same directory as the model input files or in a central location and referenced either by adding it to the PATH or by a script files. Further discussion on script files for Windows is below.

Often there is a need to run the model with no estimation. Alternative methods to run SS3 without estimating parameters are documented in the Running Without Estimation section.

As of ADMB 12.3, a new command called "-hess_step" is available to use and documented in the Using -hess_step to do additional Newton steps using the inverse Hessian

Example of DOS batch input file

One file management approach is to put ss3.exe in its own folder (example: C:\SS3_model) and to put your input files in separate folder (example: C:\My Documents \SS3_runs). Then a DOS batch file in the SS3_runs folder can be run at the command line to start ss3.exe. All output will appear in SS3_runs folder.

A DOS batch file (e.g., SS3.bat) might contain some explicit ADMB commands, some implicit commands, and some DOS commands:

```
c:\SS3_model\ss3.exe -cbs 5000000000 -gbs 50000000000 \%1 \%2 \%3 \%4
del ss.r0*
del ss.p0*
del ss.b0*
```

In this batch file, the -cbs and -gbs arguments allocate a large amount of memory for SS3 to use (you may need to edit these for your computer and SS3 configuration), and the %1, %2 etc., allows passing of command line arguments such as -nox or -nohess. You add more items to the list of % arguments as needed.

An easy way to start a command line in your current directory (SS3_runs) is to create a shortcut to the DOS command line prompt. The shortcut's target would be:

```
> %SystemRoot%\system32\cmd.exe
```

And it would start in:

```
> %CURRDIR%
```

An alternative shortcut is to have the executable within the model folder then use Ctrl+Shift+Right Click and then select either "Open powershell window here" or "Open command window here", depending upon your computer. From the command window the executable name can be typed along with additional inputs (e.g., -nohess) and the model run. If using the powershell type cmd and then hit enter prior to calling the model (ss).

Simple Batch

This first example relies upon having a set of prototype SS3 input files, where a starter file named starter.r01 can be renamed to starter.ss and then used in the SS3 run. The example also copies one of the output files, ss.std, to a new name, ss-std01.txt, to save it from being overwritten in subsequent runs. The example code should be put in a batch file, which can have any name with the .bat extension. Note that brief output from each run will be appended to cumreport.sso (see below).

```
del ss.cor
del ss.std
copy starter.r01 starter.ss
c:\admodel\ss3\ss3.exe -sdonly
copy ss.std ss-std01.txt
```

The commands could be repeated again, except the output should be copied to a different file, e.g., ss-std02.txt. This sequence can be repeated an unlimited number of times.

Complicated Batch

This second example processes 25 dat files from a different directory, each time using the same ctl and nam file. The loop index is used in the file names, and the output is searched for particular keywords to accumulate a few key results into the file SUMMARY.TXT. Comparable batch processing can be accomplished by using R or other script processing programs.

```
del summary.txt
del ss-report.txt
copy /Y runnumber.zero runnumber.ss
FOR /L \%\%i IN (1,1,25) DO (
copy /Y ..\MakeData\A1-D1-%%i.dat  Asel.dat
del ss.std
del ss.cor
del ss3.par
c:\admodel\ss3\ss3.exe
```

```
copy /Y ss3.par A1-D1-A1-%%i.par
copy /Y ss.std A1-D1-A1-%%i.std
find ''Number'' A1-D1-A1-%%i.par >> Summary.txt
find ''hessian'' ss.cor >> Summary.txt)
```

Running Without Estimation

There may be time when users will want to run the model without parameter estimation. The ADMB command -noest will not work with Stock Synthesis, as it bypasses the procedure section. There are two suggested alternative approaches to do this with SS3 and ADMB.

The first approach requires the user to change the maximum phase value in the starter.ss file to 0 then running the model via the command widow as without calculating the hessian:

```
ss3 -nohess
```

The second approach is done all through the command window using the following commands:

```
ss3 -maxfn 0 -phase 50 -nohess
```

where maxfun specifies the number of function calls and phase is the maximum phase for the model to start estimation where the number should be greater than the maximum phase for estimating parameters within the model.

However, the approaches above differ in subtle ways. First, if the maximum phase is set to 0 in the starter file the total likelihood will differ by a small amount (0.25 likelihood units) compared to the second approach which sets the maxfun and phase in the command window. This small difference is due a dummy parameter which is evaluated by the objective function when maximum phase in the starter is set to 0, resulting in a small contribution to the total likelihood of 0.25. However, all other likelihood components should not change.

The second difference between the two no estimation approaches is the reported number of "Active_count" of parameters in the Report file. If the command line approach is used (ss3 -maxfn 0 -phase 50 -nohess) then the active number of parameters will equal the number of parameters with positive phases, but because the model is started in a phase greater than the maximum phase in the model, these parameters do not move from the initial values in

the control file (or the par file). The first approach where the maximum phase is changed in the starter file will report the number of "Active_count" parameters as 0.

The final thing to consider when running a model without estimation is whether you are starting from the par file or the control file. If you start from the par file (specified in the starter file: 1=use ss3.par) then all parameters, including parameter deviations, will be fixed at the estimated values. However, if the model is not run with the par file, any parameter deviations (e.g., recruitment deviations) will not be included in the model run (a user could paste in the estimated recruitment deviations into the control file).

Generate .ss_new files

There may be times a user would like to generate the .ss_new files without running the model, with or without estimation. There are two approaches that a user can take. The first is to manually change the maxphase in the starter.ss file to -1 and running the model as normal will generate these files without running through the model dynamics (e.g., no Report file will be created). The maxphase in the starter.ss_new file will be set to -1 and will need to be manually changed back if the intent is the replace the original (i.e., starter.ss) file with the new files (i.e., starter.ss_new). The second approach is to modify the maxphase via the command line or power shell input. Calling the model using the commands:

```
ss3 -stopph -1
```

where -1 is the maximum phase for the model to run through (e.g., can be other values if a user would like to only run through a specific parameter phase). This approach will create all the new files with the starter.ss_new reflecting the original maxphase value in the starter.ss file. This approach is available in v.3.30.16 and later.

Using -hess_step to do additional Newton steps using the inverse Hessian

The optimizer in ADMB is designed to run until the maximum absolute gradient (mag) is small enough (e.g., 1e-05), after which it quits and does the uncertainty calculations. But if run for longer it cannot appreciably decrease this mag. In many cases it is interesting or advisable to get closer to the mode to confirm convergence of the model.

A new feature as of ADMB 12.3 called "-hess_step" takes Newton steps to update the MLE using the information in the Hessian calculated as $MLE_{new}=MLE-(inverse\ Hessian)*(gradient)$, where the Hessian and gradient are calculated from the original MLE. If the mag improves then this corroborates the optimizer has converged and that the negative log-likelihood surface is approximately quadratic at the

mode as assumed in the asymptotic uncertainty calculations. The downside is the high computational cost due to the extra matrix calculations.

The feature is used by optimizing normally, and then from the command line running -hess_step for defaults (recommended), -hess_step N, or -hess_step_tol eps where N and eps are the maximum number of steps to take and the tolerance (i.e., a very small number like 1e-10) after which to stop. When running the Hessian first and then the -hess_step, ADMB will prompt you to run it with -binp ss.bar.

Running Parameter Profiles

Users will often want to run profiles over specific parameter to evaluate the information in the model to estimate the parameter based on changes in the log likelihood. There are two ways this can be done.

The first option is the use functions within **r4ss** to run the profile, summarize quantities across runs, and plot the output. The **SS_profile()** function will run the profile based on function inputs, **SSgetoutput()** will read quantities from each run Report file, **SSsummarize()** will summarize key model quantities, and the **SSplotProfile()** and **PinerPlot()** functions can be used to visualize results. Additional information regarding **r4ss** can be found in the r4ss section.

The second way is to create and run a batch file to profile over parameters. This example will run a profile on natural mortality and spawner-recruitment steepness, of course. Edit the control file so that the natural mortality parameter and steepness parameter lines have the phase set to -9999. Edit starter.ss to refer to this control file and the appropriate data file.

> Create a profilevalues.ss file
> 2 # number of parameters using profile feature
> 0.16 # value for first selected parameter when runnumber equals 1
> 0.35 # value for second selected parameter when runnumber equals 1
> 0.16 # value for first selected parameter when runnumber equals 2
> 0.40 # value for second selected parameter when runnumber equals 2
> 0.18 # value for first selected parameter when runnumber equals 3
> 0.40 # value for second selected parameter when runnumber equals 3
> etc.; make it as long as you like.

Create a batch file that looks something like this. Or make it more complicated as in the example above.

> ```
> del cumreport.sso
> ```

```
copy /Y runnumber.zero runnumber.ss  % so you will start with runnumber=0
C:\SS330\ss3.exe
C:\SS330\ss3.exe
C:\SS330\ss3.exe
```

Repeat as many times as you have set up conditions in the profilevalues.ss file. The summary results will all be collected in the cumreport.sso file. Each step of the profile will have an unique run number and its output will include the values of the natural mortality and steepness parameters for that run.

Re-Starting a Run

Model runs can be restarted from a previously estimated set of parameter values. In the starter.ss file, enter a value of 1 on the first numeric input line. This will cause the model to read the file ss3.par and use these parameter values in place of the initial values in the control file. This option only works if the number of parameters to be estimated in the new run is the same as the number of parameters in the previous run because only actively estimated parameters are saved to the file ss3.par. The file ss3.par can be edited with a text editor, so values can be changed and rows can be added or deleted. However, if the resulting number of elements does not match the setup in the control file, then unpredictable results will occur. Because ss3.par is a text file, the values stored in it will not give exactly the same initial results as the run just completed. To achieve greater numerical accuracy, the model can also restart from ss.bar which is the binary version of ss3.par. In order to do this, the user must make the change described above to the starter.ss file and must also enter -binp ss.bar as one of the command line options.

Optional Output Subfolders

As of v.3.30.19, users can optionally send .sso and .ss_new extension files to subfolders. To send files with a .sso extension to a subfolder within the model folder, create a subfolder called sso before running the model. To send files with a .ss_new extension to a separate subfolder, create a folder called ssnew before running the model.

Putting Stock Synthesis in your PATH

Instead of copying the SS3 executable to each model folder, SS3 can be put in your system path, which is a list of folders that your operating system looks in whenever you type the name of a program on the command line. This approach saves on storage space since the SS3 binary (i.e., the SS3 executable or exe) is about 2.2 MB and having it located in each folder can be prohibitive in a large-scale simulation testing study. Even if you are not running a large simulation study, putting SS3 in your path may still be convenient, as you can use the same executable on many models, there is no need to specify a full file path to

the executable each time you run a model, and no need to create a batch file that refers to the executable's location.

For Unix (OS X and Linux)

To check if SS3 is in your path, assuming the binary is named SS3: open a Terminal window and type `which SS3` and hit enter. If you get nothing returned, then SS3 (named SS3 or SS3.exe) is not in your path. The easiest way to fix this is to move the SS3 binary to a folder that's already in your path. To find existing path folders type `echo $PATH` in the terminal and hit enter. Now move the SS3 binary to one of these folders.

For example, in a Terminal window type:

```
sudo cp ~/Downloads/SS3 /usr/bin/
```

to move an binary called SS3 from the Downloads folder to `/usr/bin`. You will need to use `sudo` and enter your password after to have permission to move a file to a folder like `/usr/bin/`, because doing so edits the system for other users also.

Also note that you may need to add executable permissions to the SS3 binary after downloading it. You can do that by switching to the folder where you placed the binary (`cd /usr/bin/` if you followed the instructions above), and running the command:

```
sudo chmod +x SS3
```

Check that SS3 is now executable and in your path:

```
which SS3
```

If you followed the instructions above, you will see the following line returned:

```
/usr/bin/SS3
```

If you have previously modified your path to add a non-standard location for the SS3 binary, you may need to also tell R about the new path. The path that R sees may not include additional paths that you have added through a configuration file like `.bash_profile`. If needed, you can add to the path that R sees by including a line like this in your `.Rprofile` file (`.Rprofile` is an invisible file in your home directory).

```
Sys.setenv(PATH=paste(Sys.getenv(''PATH''),''/my/folder'',sep='''':''))
```

For Windows

To check if SS3 is in your path for Windows, open a DOS prompt (either Command Prompt or Powershell should work) and type `SS3 -?` and hit enter. If the prompt returns a message like `SS3 is not recognized...`, then SS3 is not in your path (assuming the SS3 executable is called SS3.exe).

To add the SS3 binary file to your path, follow these steps:

1. Find the correct version of the SS3.exe binary on your computer (or download from the SS3 releases).

2. Move to and note the folder location. E.g., `C:/SS3/`

3. Click on the start menu and type `environment`

4. Choose `Edit environment variables for your account` under Control Panel

5. Click on `PATH` if it exists, create it if does not exist

6. Choose 'PATH' and click edit

7. In the `Edit User Variable` window add to the end of the `Variable value` section a semicolon and the SS3 folder location you recorded earlier. E.g., `;C:/SS3`. Do not overwrite what was previously in the `PATH` variable.

8. Restart your computer

9. Go back to the DOS prompt and try typing `SS3 -?` and hitting return again.

Running Stock Synthesis from R

Use `system("path/to/ss3")` to run Stock Synthesis from within the R console, where `path/to/ss3` is the path to and name of the Stock Synthesis binary.

Alternatively, use the function `run` from the `r4ss` package within the R console:

```
# run model, in directory folder_1, using the SS3 executable
# named ss3 that is in the path.
r4ss::run(dir = ''folder_1'',
          exe = ''ss3'')
```

Running SS3 from within R may be desirable for setting up simulations where many runs of SS3 models are required (e.g., `ss3sim`) or if `r4ss` is already used to read model output.

The Stock Synthesis GUI (SSI)

Stock Synthesis Interface (SSI or the SS3 GUI) provides an interface for loading, editing, and running model files, and also can link to r4ss to generate plots. Note that SSI is not maintained for Stock Synthesis versions after v.3.30.21.

The Stock Assessment Continuum Tool

The Stock Assessment Continuum Tool (previously known as the Stock Synthesis Data-limited Tool) is a Shiny-based application that uses SS3 as the flexible framework to apply a variety of model types depending on the available data (catch time-series, age composition, length composition, abundance index data). It is meant to make SS3 accessible to users, open up many features and tools associated with Stock Synthesis, provide an easy way to enter data in the model, and make model specification and uncertainty exploration easier.

Debugging Tips

When input files are causing the program to crash or fail to produce sensible results, there are a few steps that can be taken to diagnose the problem. Before trying the steps below, examine the echoinput.sso file. It is highly annotated, so you should be able to see if the model is interpreting your input files as you intended. Additionally, users should check the warning.sso file when attempting to debug a non-running model.

1. Set the turn_off_phase switch to 0 in the starter.ss file. This will cause the mode to not attempt to adjust any parameters and simply converges a dummy parameter. It will still produce a Report.sso file, which can be examined to see what has been calculated from the initial parameter values.

2. Turn the verbosity level to 2 in the starter.ss file. This will cause the program to display the value of each likelihood component to the screen on each iteration. So it the program is creating an illegal computation (e.g., divide by zero), it may show you which likelihood component contains the problematic calculation. If the program is producing a Report.sso file, you may then see which observation is causing the illegal calculation.

3. Run the program with the command ss3 »SSpipe.txt. This will cause all screen display to go to the specified text file (note, delete this file before running because it

will be appended to). Examination of this file will show detailed statements produced during the reading and preprocessing of input files.

4. If the model fails to achieve a proper Hessian it exits without writing the detailed outputs in the FINAL_SECTION. If this happens, you can do a run with the -nohess option so you can view the Report.sso to attempt to diagnose the problem.

5. If the problem is with reading one or more of the input files, please note that certain Mac line endings cannot be read by the model (although this is a rare occurrence). Be sure to save the text files with Windows or Linux style line endings so that the executable can parse them.

Keyboard Tips

Typing "N" during a run will cause ADMB to immediately advance to the next phase of estimation.

Typing "Q" during a run will cause ADMB to immediately go to the final phase. This bypasses estimation of the Hessian and will produce all of the model outputs, which are coded in the FINAL_SECTION.

Running MCMC

Running SS3 with MCMC can be done through command line options using the default ADMB MCMC algorithm (described below). Another possibility is using the R package adnuts. See the adnuts vignette for more information. The MCMC guide for ADMB provides the most comprehensive guidance available for using MCMC with ADMB models (such as SS3). Additional guidance is available in (Monnahan et al., 2019).

Running SS3 with MCMC (instead of maximum likelihood estimation) provides maximum posterior density estimates, report file, Hessian matrix and the .cor file. Parameters stuck on bounds which will degrade efficiency of MCMC implementation. Two commands are needed to obtain the model results:

Run SS3 with arguments -mcmc xxxx -mcsave yyyy

- Where: xxxx is the number of iterations for the chain, and yyyy is the thinning interval (1000 is a good place to start).

- MCMC chain starts at the MPD values.

- Recommended: Remove existing .psv files in run directory to generate a new chain.

- Recommended: Before running, set the run detail switch in starter file to 0 to limit printing to the screen; reporting to screen will slow MCMC progress.

- Optional: Add `-nohess` to use the existing Hessian file without re-estimating.

- Optional: To start the MCMC chain from specific values change the par file: run the model with estimation, adjust the par file to the values that the chain should start from, change within the starter file for the model to begin from the par file, and call the MCMC function using `ss3 -mcmc xxxx - mcsave yyyy -nohess -noest`.

Run SS3 with argument -mceval to get more summaries

- This generates the posterior output files.

- Optional: Modify starter file entries to add a burn-in and thinning interval above and beyond the ADMB thinning interval applied at run time.

- Recommended: MCMC always begins with the maximum posterior density values and so a burn-in >0 should always be used.

- This step can be repeated for alternate forecast options (e.g., catch levels) without repeating step 2.

Note that when the model is switched to MCMC or MCEVAL mode, all the bias adjustment factors become 1.0 for any years with recruitment deviations. A report file is not created after completing MCMC because it would show values based only on the last MCMC step.

Output Files

Custom Reporting

Additional user control for what is included in the Report.sso file was added in v.3.30.16. This approach allows for full customizing of what is printed to the Report file by selecting custom reporting (option = 3) in the starter file where specific items now can be included or excluded depending upon a list passed to SS3 from the starter file. The numbering system for each item in the Report file is as follows:

Num.	Report Item	Num.	Report Item
1	DEFINITIONS	31	LEN SELEX
2	LIKELIHOOD	32	AGE SELEX
3	Input Variance Adjustment	33	ENVIRONMENTAL DATA
4	Parm devs detail	34	TAG Recapture
5	PARAMETERS	35	NUMBERS-AT-AGE
6	DERIVED QUANTITIES	36	BIOMASS-AT-AGE
7	MGparm By Year after adjustments	37	NUMBERS-AT-LENGTH
8	selparm(Size) By Year after adjustments	38	BIOMASS-AT-LENGTH
9	selparm(Age) By Year after adjustments	39	F-AT-AGE
10	RECRUITMENT DIST	40	CATCH-AT-AGE
11	MORPH INDEXING	41	DISCARD-AT-AGE
12	SIZEFREQ TRANSLATION	42	BIOLOGY
13	MOVEMENT	43	Natural Mortality
14	EXPLOITATION	44	AGE SPECIFIC K
15	CATCH	45	Growth Parameters
16	TIME SERIES	46	Seas Effects
17	SPR SERIES	47	Biology at age in endyr
18	Kobe Plot	48	MEAN BODY WT(Begin)
19	SPAWN RECRUIT	49	MEAN SIZE TIMESERIES
20	SPAWN RECR CURVE	50	AGE LENGTH KEY
21	INDEX 1	51	AGE AGE KEY
22	INDEX 2	52	COMPOSITION DATABASE
23	INDEX 3	53	SELEX database
24	DISCARD SPECIFICATION	54	SPR/YPR Profile
25	DISCARD OUTPUT	55	GLOBAL MSY
26	MEAN BODY WT OUTPUT	56	SS_summary.sso

27	FIT LEN COMPS	57	rebuilder.sso
28	FIT AGE COMPS	58	SIStable.sso
29	FIT SIZE COMPS	59	Dynamic Bzero
30	OVERALL COMPS	60	wtatage.ss_new

Standard ADMB output files

Standard ADMB files are created by SS3. These are:

ss3.par (previously ss.par) - This file has the final parameter values. They are listed in the order they are declared in SS3. This file can be read back into SS3 to restart a run with these values (see Running Stock Synthesis for more info).

ss.std - This file has the parameter values and their estimated standard deviation for those parameters that were active during the model run. It also contains the derived quantities declared as standard deviation report variables. All of this information is also report in the covar.sso. Also, the parameter section of Report.sso lists all the parameters with their SS3 generated names, denotes which were active in the reported run, displays the parameter standard deviations, then displays the derived quantities with their standard deviations.

ss.rep - This report file is created between phases so, unlike Report.sso, will be created even if the hessian fails. It does not contain as much output as shown in Report.sso.

ss.cor - This is the standard ADMB report for parameter and standard deviation report correlations. It is in matrix form and challenging to interpret. This same information is reported in covar.sso.

Stock Synthesis Summary

The ss_summary.sso file (available for versions 3.30.08.03 and later) is designed to put key model outputs all in one concise place. It is organized as a list. At the top of the file are descriptors, followed by the 1) likelihoods for each component, 2) parameters and their standard errors, and 3) derived quantities and their standard errors. Total biomass, summary biomass, and catch were added to the quantities reported in this file in v.3.30.11 and later.

Before v.3.30.17, TotBio and SmryBio did not always match values reported in columns of the TIME_SERIES table of Report.sso. The report file should be used instead of ss_summary.sso for correct calculation of these quantities before v.3.30.17. Care should be

taken when using the TotBio and SmryBio if the model configuration has recruitment after January 1 or in a later season, as TotBio and SmryBio quantities are always calculated on January 1. Consult the detailed age-, area-, and season-specific tables in report.sso for calculations done at times other than January 1.

SIS table

The SIS_table.sso is deprecated as of v.3.30.17. Please use the r4ss function `get_SIS_info()` instead.

The SIS_table.sso file contains model output formatted for reading into the NMFS Species Information System (SIS). This file includes an assessment summary for categories of information (abundance, recruitment, spawners, catch estimates) that are input into the SIS database. A time-series of estimated quantities which aggregates estimates across multiple areas and seasons are provided to summarize model results. Access to the SIS database is granted to all NOAA employees.

Derived Quantities

Before listing the derived quantities reported to the standard deviation report, there are a couple of topics that deserve further explanation.

Virgin Spawning Biomass vs Unfished Spawning Biomass

Unfished is the condition for which reference points (benchmark) are calculated. Virgin Spawning Biomass (B0) is the initial condition on which the start of the time-series depends. If biology or spawner-recruitment parameters are time-varying, then the benchmark year input in the forecast file tells the model which years to average in order to calculate "unfished". In this case, virgin recruitment and/or the virgin spawning biomass will differ from their unfished counterparts. Virgin recruitment and spawning biomass are reported in the mgmt_quant portion of the sd_report and are now labeled as "unfished" for clarity. Note that if ln(R0) is time-varying, then this will cause unfished to differ from virgin. However, if regime shift parameter is time-varying, then unfished will remain the same as virgin because the regime shift is treated as a temporary offset from virgin. Virgin spawning biomass is denoted as SPB_virgin and spawning biomass unfished is denoted as SPB_unf in the report file.

Virgin Spawning Biomass (B0) is used in four ways within SS3:

1. Anchor for the spawner-recruitment relationship as virgin spawning biomass.

2. Basis for the initial equilibrium abundance.

3. Basis against which annual depletion is calculated.

4. Benchmark calculations.

However, if there is time-varying biology, then the 4th usage can have a different B0 calculation compared to the other usages.

Metric for Fishing Mortality

A generic single metric of annual fishing mortality is difficult to define in a generalized model that admits multiple areas, multiple biological cohorts, dome-shaped selectivity in size and age for each of many fleets. Several separate indices are provided and others could be calculated by a user from the detailed information in Report.sso.

Equilibrium SPR

This index focuses on the effect of fishing on the spawning potential of the stock. It is calculated as the ratio of the equilibrium reproductive output per recruit that would occur with the current year's F intensities and biology, to the equilibrium reproductive output per recruit that would occur with the current year's biology and no fishing. Thus it internalizes all seasonality, movement, weird selectivity patterns, and other factors. Because this index moves in the opposite direction than F intensity itself, it is usually reported as 1-SPR. A benefit of this index is that it is a direct measure of common proxies used for F_{MSY}, such as $F_{40\%}$. A shortcoming of this index is that it does not directly demonstrate the fraction of the stock that is caught each year. The SPR value is also calculated in the benchmarks (see below).

The derived quantities report shows an annual SPR statistic. The options, as specified in the starter.ss file, are:

- 0 = skip

- 1 = (1-SPR)/(1-SPR_{TGT})

- 2 = (1-SPR)/(1-SPR_{MSY})

- 3 = (1-SPR)/(1-$SPR_{Btarget}$)

- 4 = raw SPR

The SPR approach to measuring fishing intensity was implemented because the concept of a single annual F does not exist in SS3 because F varies by age, sex, and growth morph and season and area. There is no single F value that is applied to all ages unless you create a very simple model setup with knife-edge selectivity. So, what you see in the options are

various ways to calculate annual fishing intensity. They can be broken down into three categories. One is exploitation rate calculated simply as total catch divided by biomass from a defined age range. Another is SPR, which is a single measure of the equilibrium effect of fishing according to the F. The third category are various ways to calculate an average F. Some measures of fishing intensity will be misleading if applied inappropriately. For example, the sum of the apical F's will be misleading if different fleets have very different selectivities or, worse, if they occur in different areas. The F=Z-M approach to getting fishing intensity is a way to have a single F that represents a number's weighted value across multiple areas, sexes, morphs, ages. An important distinction is that the exploitation rate and F-based approaches directly relate to the fraction of the population removed each year by fishing; whereas the SPR approach represents the cumulative effect of fishing so it's equivalent in F-space depends on M.

F std

This index provides a direct measure of fishing mortality. The options are:

- $0 = $ skip

- $1 = $ exploitation(Bio)

- $2 = $ exploitation(Num)

- $3 = $ sum(Frates)

The exploitation rates are calculated as the ratio of the total annual catch (in either biomass or numbers as specified) to the summary biomass or summary numbers on January 1. The sum of the F rates is simply the sum of all the apical Fs. This makes sense if the F method is in terms of instantaneous F (not Pope's approximation) and if there are not fleets with widely different size/age at peak selectivity, and if there is no seasonality, and especially if there is only one area. In the derived quantities, there is an annual statistic that is the ratio of the can be annual F_std value to the corresponding benchmark statistic. The available options for the denominator are:

- $0 = $ raw

- $1 = $ F/F$_{\text{SPR}}$

- $2 = $ F/F$_{\text{MSY}}$

- $3 = $ F/F$_{\text{Btarget}}$

- $>= 11$ A new option to allow for the calculation of a multi-year trailing average in F was implemented in v.3.30.16. This option is triggered by appending the number of

years to calculate the average across where an input of 1 or 11 would result in the $\text{SPR}_{\text{target}}$ with no changes. Alternatively a value of 21 would calculate F as $\text{SPR}_{\text{target}}$ with a 2-year trailing average.

F-at-Age

Because the annual F is so difficult to interpret as a sum of individual F components, an indirect calculation of F-at-age is reported at the end of the report.sso file. This section of the report calculates Z-at-age simply as $ln(N_{a+1,t+1}/N_{a,t})$. This is done on an annual basis and summed over all areas. It is done once using the fishing intensities as estimated (to get Z), and once with the F intensities set to 0.0 to get M-at-age. This latter sequence also provides a measure of dynamic Bzero. The user can then subtract the table of M-at-age/year from the table of Z-at-age/year to get a table of F-at-age/year. From this apical F, average F over a range of ages, or other user-desired statistics could be calculated. Further work within SS3 with this table of values is anticipated.

MSY and other Benchmark Items

The following quantities are included in the sdreport vector mgmt_quantities, so obtain estimates of variance. Some additional quantities can be found in the benchmarks section of the forecast_report.sso.

Benchmark Item	Description
SSB_Unfished	Unfished reproductive potential (SSB is commonly female mature spawning biomass).
TotBio_Unfished	Total age 0+ biomass on January 1.
SmryBio_Unfished	Biomass for ages at or above the summary age on January 1.
Recr_Unfished	Unfished recruitment.
SSB_Btgt	SSB at user specified SSB target.
SPR_Btgt	Spawner potential ratio (SPR) at F intensity that produces user specified SSB target.
Fstd_Btgt	F statistic at F intensity that produces user specified SSB target.
TotYield_Btgt	Total yield at F intensity that produces user specified SSB target.
SSB_SPRtgt	SSB at user specified SPR target (but taking into account the spawner-recruitment relationship).

Benchmark Item	Description
Fstd_SPRtgt	F intensity that produces user specified SPR target.
TotYield_SPRtgt	Total yield at F intensity that produces user specified SPR target.
SSB_MSY	SSB at F intensity that is associated with MSY; this F intensity may be directly calculated to produce MSY, or can be mapped to F_SPR or F_Btgt.
SPR_MSY	Spawner potential ratio (SPR) at F intensity associated with MSY.
Fstd_MSY	F statistic at F intensity associated with MSY.
TotYield_MSY	Total yield (biomass) at MSY.
RetYield_MSY	Retained yield (biomass) at MSY.

Brief cumulative output

Cum_Report.sso: contains a brief version of the run output, which is appended to current content of file so results of several runs can be collected together. This is especially useful when a batch of runs is being processed. Unless this file is deleted, it will contain a cumulative record of all runs done in that subdirectory. The first column contains the run number.

Bootstrap Data Files

It is possible to create bootstrap data files for SS3 where an internal parametric bootstrap function generates a simulated data set by parametric bootstrap sampling the expected values given the input observation error. Starting in v.3.30.19, bootstrap data files are output separated in single numbered files (e.g., data_boot_001.ss). In version prior to v.3.30.19 a single file called data.ss_new was output that contained multiple sections: the original data echoed out, the expected data values based on the model fit, and then subsequent bootstrap data files.

Specifying the number of bootstrap data files has remained the same across model versions. Creating bootstrap data files is specified in the starter file via the "Number of datafiles to produce" line where a value of 3 or greater will create three files: the original data file, data_echo.ss_new, a data file with the model expected values, data_expval.ss, and single

bootstrap data file, data_boot_001.ss. The first output provides the unaltered input data file (with annotations added). The second provides the expected values for only the data elements used in the model run. The third and subsequent outputs provide parametric bootstraps around the expected values.

The bootstrapping procedure within SS3 is done via the following steps:

- Expected values of all input data are calculated (these are also used in the likelihood which compares observed to expected values for all data). The calculation of these expected values is described in detail under the "Observation Model" section of the appendix to Methot and Wetzel (2013).

- Parametric bootstrap data are calculated for each observation by sampling from a probability distribution corresponding to the likelihood for that data type using the expected values noted above. Examples of how this happens include the following:

  - Indices of abundance are sampled from the distribution used in the estimation model (as set in the "Errtype" column of the index configuration, most commonly lognormal but could be normal or T-distribution). The variability of the distribution from which the random sample is drawn is based on the combination of the input uncertainty and any estimated "Extra SD" parameter and any "add_to_survey_CV" value included under the input variance adjustments factors.

  - Length and age compositions are sampled from multinomial distributions with expected proportions in each bin based on the expected values and sample size equal to the adjusted input sample size (input sample size multiplied by any inputs for "mult_by_lencomp_N" or "mult_by_agecomp_N" under the input variance adjustments factors).

  - Discard data (fractions or absolute amounts) are generated from the chosen distribution (T-distribution, normal, log-normal, truncated-normal).

  - Tagging data is generated using a negative binomial distribution to get the total number of recaptures for each tag group and a multinomial distribution to allocate those recaptures among fleets.

Given this, there are some assumptions implicit in the bootstrapping procedure (as implemented as of v.3.30.17) that users should be aware of:

- This procedure is strictly an observation error approach (i.e., process error in recruitment or any other time-varying parameter is not added). For simulation analyses, a common approach has been to input a new time series of recruitment deviations for each bootstrap data set.

- The sample size for conditional age-at-length data matches the inputs for each length bin. If stratified sampling is used, this may be appropriate, but if the ages represent a random subset of the selected population, this may result in less variability than if the associated length distribution were resampled.

- Currently, the aging error matrix is multiplied by the expected distribution of proportions at age, while the more correct order of operations would be to sample true ages, and then sample the observed age including aging error (it is possible these are mathematically identical).

- Often there is need to explore the removal (not include in the model fitting) of specific years in a data set which can be done by specifying a negative fleet number. If bootstrapping a data file, note that specifying a negative fleet in the data inputs for indices, length composition, or age composition will include the "observation" in the model (hence generating predicted values and bootstrap data sets for the data), but not in the negative log likelihood. The "observation values" used with negative fleet do not influence the predicted values, except when using tail compression with length or age composition. Non-zero values greater than the minimum tail compression should be used for the observation values when tail compression is being used, as using zeros or values smaller than the minimum tail compression can cause the predicted values to be reported as zero and shift predictions to other bins.

- As of v.3.30.15, age and length composition data that use the Dirichlet-Multinomial distribution in the model are generated using the Dirichlet-Multinomial in bootstrap data sets.

Forecast and Reference Points (Forecast-report.sso)

The Forecast-report file contains output of fishery reference points and forecasts. It is designed to meet the needs of the Pacific Fishery Management Council's Groundfish Fishery Management Plan, but it should be quite feasible to develop other regionally specific variants of this output.

The vector of forecast recruitment deviations is estimated during an additional model estimation phase. This vector includes any years after the end of the recruitment deviation time series and before or at the end year. When this vector starts before the ending year of the time series, then the estimates of these recruitments will be influenced by the data in these final years. This is problematic, because the original reason for not estimating these recruitments at the end of the time series was the poor signal/noise ratio in the available data. It is not that these data are worse than data from earlier in the time series, but the low amount of data accumulated for each cohort allows an individual datum to dominate the model's fit. Thus, an additional control is provided so that forecast recruitment

deviations during these years can receive an extra weighting in order to counter-balance the influence of noisy data at the end of the time series.

An additional control is provided for the fraction of the log-bias adjustment to apply to the forecast recruitments. Recall that R is the expected mean level of recruitment for a particular year as specified by the spawner-recruitment curve and R' is the geometric mean recruitment level calculated by discounting R with the log-bias correction factor $e - 0.5s^2$. Thus a lognormal distribution of recruitment deviations centered on R' will produce a mean level of recruitment equal to R. During the modeled time series, the virgin recruitment level and any recruitments prior to the first year of recruitment deviations are set at the level of R, and the lognormal recruitment deviations are centered on the R' level. For the forecast recruitments, the fraction control can be set to 1.0 so that 100% of the log-bias correction is applied and the forecast recruitment deviations will be based on the R' level. This is certainly the configuration to use when the model is in MCMC mode. Setting the fraction to 0.0 during maximum likelihood forecasts would center the recruitment deviations, which all have a value of 0.0 in maximum likelihood mode, on R. Thus would provide a mean forecast that would be more comparable to the mean of the ensemble of forecasts produced in MCMC mode. Further work on this topic is underway.

Note:

- Cohorts continue growing according to their specific growth parameters in the forecast period rather than staying static at the end year values.

- Environmental data entered for future years can be used to adjust expected recruitment levels. However, environmental data will not affect growth or selectivity parameters in the forecast.

The top of the Forecast-report file shows the search for F_{SPR} and the search for F_{MSY}, allowing the user to verify convergence. Note: if the STD file shows aberrant results, such as all the standard deviations being the same value for all recruitments, then check the F_{MSY} search for convergence. The F_{MSY} can be calculated, or set equal to one of the other F reference points per the selection made in starter.ss.

Main Output File, Report.sso

This is the primary output file. Its major sections (as of v.3.30.16) are listed below.

The sections of the output file are:

- SS3 version number with date compiled. Time and date of model run. This info appears at the top of all output files.

- Comments

– Input file lines starting with #C are echoed here.

- Keywords

  – List of keywords used in searching for output sections.

- Definitions

  – List of definitions (e.g., fleet names, model start year) assigned in the data and control files.

- Likelihood

  – Final values of the negative log(likelihood) are presented.

- Input Variance Adjustments

  – The matrix of input variance adjustments is output here because these values affect the log likelihood calculations.

- Parm deviations detail

  – Details about parameter deviations, if used in the model. Will be missing if no parameter devs were used.

- Parameters

  – The parameters are listed here. For the estimated parameters, the display shows: Num (count of parameters), Label (as internally generated by SS3), Value, Active_Cnt, Phase, Min, Max, Init, Prior, Prior_type, Prior_SD, Prior_Like, Parm_StD (standard deviation of parameter as calculated from inverse Hessian), Status (e.g., near bound), and Pr_atMin (value of prior penalty if parameter was near bound). The Active_Cnt entry is a count of the parameters in the same order they appear in the ss.cor file.

- Derived Quantities

  – This section starts by showing the options selected from the starter.ss and forecast.ss input files:

    * SPR ratio basis

    * F report basis

    * B ratio denominator

196

Then the time series of output, with standard deviation of estimates, are produced with internally generated labels. Note that these time series extend through the forecast era. The order of the output is: spawning biomass, recruitment, SPRratio, Fratio, Bratio, management quantities, forecast catch (as a target level), forecast catch as a limit level (OFL), Selex_std, Grow_std, NatAge_std. For the three "ratio" quantities, there is an additional column of output showing a Z-score calculation of the probability that the ratio differs from 1.0. The "management quantities" section is designed to meet the terms of reference for west coast groundfish assessments; other formats could be made available upon request. The standard deviation quantities at the end are set up according to specifications at the end of the control input file. In some cases, a user may specify that no derived quantity output of a certain type be produced. In those cases, SS3 substitutes a repeat output of the virgin spawning biomass so that vectors of null length are not created.

- Mortality and growth parameters by year after adjustments

    – This block shows the time series of mortality and growth parameters by year after adjustments by environmental links, blocks and deviations.

- Selectivity parameters (size) by year after adjustments

    – This block shows the size selectivity parameters, after adjustment, for each year in which a change occurs.

- Selectivity parameters (age) by year after adjustments

    – This block shows the age selectivity parameters, after adjustment, for each year in which a change occurs.

- Recruitment Distribution

    – This block shows the distribution of recruitment across growth patterns, sexes, birth seasons, and areas in the end year of the model.

- Growth Morph Indexing

    – This block shows the internal index values for various quantities. It can be a useful reference for complex model setups. The vocabulary is: Bio_Pattern refers to a collection of cohorts with the same defined growth and natural mortality parameters; sex is the next main index. If recruitment occurs in multiple seasons, then birth season is the index for that factor. The index labeled "Platoon" is used as a continuous index across all the other factor-specific indices. If sub-platoons are used, they are nested within the Bio_Pattern x Sex x Birth Season platoon. However, some of the output tables use the column label "platoon" as a continuous index across platoons and

sub-platoons. Note that there is no index here for area. Each of the cohorts is distributed across areas and they retain their biological characteristics as they move among areas.

- Size Frequency Translation

  – If the generalized size frequency approach is used, this block shows the translation probabilities between population length bins and the units of the defined size frequency method. If the method uses body weight as the accumulator, then output is in corresponding units.

- Movement

  – This block shows movement rate between areas in a multi-area model.

- Exploitation

  – This block shows the time series of the selected F_std unit and the F multiplier for each fleet in terms of harvest rate (if Pope's approximation is used) or fully selected F.

- Catch

- Time Series

- SPR Series

- Kobe Plot

  – Reports output in a table needed to create a Kobe Plot.

- Spawn Recruit Parameters and Table

  – Extensive information on Spawn-recruit parameter values and derived quantities.

- Spawn Recruit Curve

  – A table containing information to recreate the spawn-recruit curve.

- Index 1

- Index 2

  – This section reports the observed and expected values for each index. All are reported in one list with index number included as a selection field. At the

198

bottom of this section, the root mean squared error of the fit to each index is compared to the mean input error level to assist the user in gauging the goodness-of-fit and potentially adjusting the input level of imprecision.

- Index 3

  – This section shows the parameter number assigned to each parameter used in this section.

- Discard Specification

- Discard Output

  – This is the list of observed and expected values for the amount (or fraction) discard.

- Mean Body Wt

  – This is the list of observed and expected values for the mean body weight.

- Fit Len Comps

  – This is the list of the goodness of fit to the length compositions. The input and output levels of effective sample size are shown as a guide to adjusting the input levels to better match the model's ability to replicate these observations.

- Fit Age Comps

  – This has the same format as the length composition section.

- Fit Size Comps

  – This has the same format as the length composition section and is used for the generalized size composition summary.

- Overall Comps

- Len Selex

  – Here is the length selectivity and other length specific quantities for each fishery and survey.

- Age Selex

  – Here is reported the time series of age selectivity and other age-related quantities for each fishery and survey. Some are directly computed in terms of

age, and others are derived from the combination of a length-based factor and the distribution of size-at-age.

- Environmental Data

  - The input values of environmental data are echoed here. Density-dependence can be used by linking to population quantities that have already been calculated at the start of the year. These include summary biomass, spawning biomass, and recruitment deviations. These three quantities are mapped into the -1, -2, and -3 columns of the environmental data matrix where they can be used as if there were environmental data input.

- Tag Recapture Information

- Numbers at Age

  - The output (in thousands of fish) is shown for each cohort tracked in the model.

- Biomass at Age

- Numbers at Length

  - The output is shown for each cohort tracked in the model.

- Biomass at Length

- F at Age

- Catch at Age

  - The output is shown for each fleet. It is not necessary to show by area because each fleet operates in only one area.

- Discard at Age

- Biology

  - The first biology section shows the length-specific quantities in the ending year of the time series only. The derived quantity spawn is the product of female body weight, maturity and fecundity per weight. The second section shows natural mortality.

- Natural Mortality

- Age-specific K

- Growth Parameters

  – This section shows the growth parameters, and associated derived quantities, for each year in which a change is estimated.

- Seasonal Effects

- Biology at Age

  – This section shows derived size-at-age and other quantities. As of v.3.30.21 sex ratio is reported by area in this output table.

- Mean Body Wt (begin)

  – This section reports the time series of mean body weight for each platoon. Values shown are for the beginning of each season of each year.

- Mean Size Time series

  – This section shows the time series of mean length-at-age for each platoon. At the bottom is the average mean size as the weighted average across all platoons for each sex.

- Age Length Key

  – This is reported for the midpoint of each season in the ending year.

- Age Age Key

  – This is the calculated distribution of observed ages for each true age for each of the defined ageing keys.

- Composition Database

  – Contains the length composition, age composition, and mean size-at-age observed and expected values. It is arranged in a database format, rather than an array of vectors.

- Selectivity Database

  – This section contains the selectivities organized as a database, rather than as a set of vectors.

- SPR/YPR Profile

- Global MSY Report

- Dynamic Bzero Report

Using R To View Model Output (r4ss)

The R package `r4ss` includes tools for summarizing and plotting results, manipulating files, visualizing model parameterizations, and other tasks. Currently, information about `r4ss` can be found on Github. The software package is under continuous development to maintain compatibility with new versions of SS3 and to improve functionality.

The latest version of `r4ss` can be installed directly from GitHub at any time via the `remotes` package in R with the following commands:

```
> install.packages(''remotes'')
> remotes::install_github(''r4ss/r4ss'')
```

Once the `r4ss` package is installed, it can be loaded:

```
> library(r4ss)
```

Two of the most commonly used functions for model diagnostics are `SS_output` and `SS_plots`. After running a model using SS3, the output files including Report.sso can be read into R using the `SS_output` function which stores quantities in a list with named objects. This list can then be passed to the `SS_plots` function, which creates a series of over 100 plots that are useful for visualizing output such as model fits to the data. For example, plots can be created using model output available in the directory "C:/myfiles/mymodels/myrun":

```
> base.model <- SS_output(''C:/myfiles/mymodels/myrun'')
> SS_plots(base.model)
```

The core functions available in `r4ss` include:

Core Functions	
SS_output	A function to create a list object for the output from Stock Synthesis
SS_plots	Plot many quantities related to output from Stock Synthesis
Download the SS3 Executable:	

203

get_ss3_exe	Download the latest version or a specified version of the SS3 executable

Model comparisons and other diagnostics:

SSsummarize	Read output from multiple SS3 models
SStableComparison	Make table comparing quantities across models
SSplotComparison	Plot output from multiple SS3 models
SSplotPars	Plot distributions of priors, posteriors, and estimates
SS_profile	Run likelihood parameter profiles
SSplotProfile	Plot likelihood profile results
PinerPlot	Plot fleet-specific contributions to likelihood profile
SS_RunJitter	Run multiple model jitters to determine best model fit
SS_doRetro	Run retrospective analysis
SSmohnsrho	Calculate Mohn's Rho values
SSplotRetroRecruits	Make retrospective pattern of recruitment estimates (a.k.a. squid plot) as seen in Pacific hake assessments
SS_fitbiasramp	Estimate bias adjustment for recruitment deviates

File manipulation for inputs:

SS_readdat	Read data file
SS_readctl	Read control file
SS_readforecast	Read forecast file
SS_readstarter	Read starter file
SS_readwtatage	Read weight-at-age file
SS_writedat	Write data file
SS_writectl	Write control file
SS_writeforecast	Write forecast file
SS_writestarter	Write starter file
SS_writewtatage	Write weight-at-age file

SS_makedatlist	Make a list for SS3 data
SS_parlines	Get parameter lines from SS3 control file
SS_changepars	Change parameters in the control file
SSmakeMmatrix	Create inputs for entering a matrix of natural mortality by age and year
SS_profile	Run a likelihood profile in SS3 (incomplete)
NegLogInt_Fn	Calculated variances of time-varying parameters using SS3 implementation of the Laplace Approximation

File manipulations for outputs:

SS_recdevs	Insert a vector of recruitment deviations into the control file

Advanced Stock Synthesis Configuration Settings and Advice

Using Time-Varying Parameters

Time-Varying Parameters

Starting in v.3.30, mortality-growth, some stock-recruitment, catchability, and selectivity base parameters can be time varying. Note that as of v.3.30.16, time-varying parameters cannot be used with tagging parameters. There are four ways a parameter can be time-varying in SS3:

1. Environmental or Density dependent Linkages: Links the base parameter with environmental data or a model derived quantity.

2. Parameter deviations: Creates annual deviations from the base parameter during a user-specified range of years.

3. Time blocks: The base parameter is changed during a "block" (or "blocks") of time (i.e., one or more consecutive years) as specified by the user.

4. Trends: A trend (shape: cumulative normal distribution function) is applied to the parameter. Trends are specified using the same input column as time blocks, but with different codes. This means that trends and time blocks cannot be used simultaneously for the same base parameter.

Environmental and density dependent linkages, parameter deviations, and either time blocks or trends can be applied to the same base parameter. The model processes each time-varying parameter specification (first time blocks and trends, then environmental linkages, then parameter deviations) and creates a time-series of intermediate values that are used as the model subsequently loops through years.

Specification of Time-Varying Parameters: Long Parameter Lines

Time-varying specifications for a parameter are invoked using elements 8 - 14 in the long parameter line setup. Each element and the options for selection related to time-varying parameters are as described below.

- Environmental or Density Dependent Link and variable (env_var&link; element 8)

  – The environmental or density dependent link and variable input is two inputs specified using a single three digit number. The hundreds place contains the
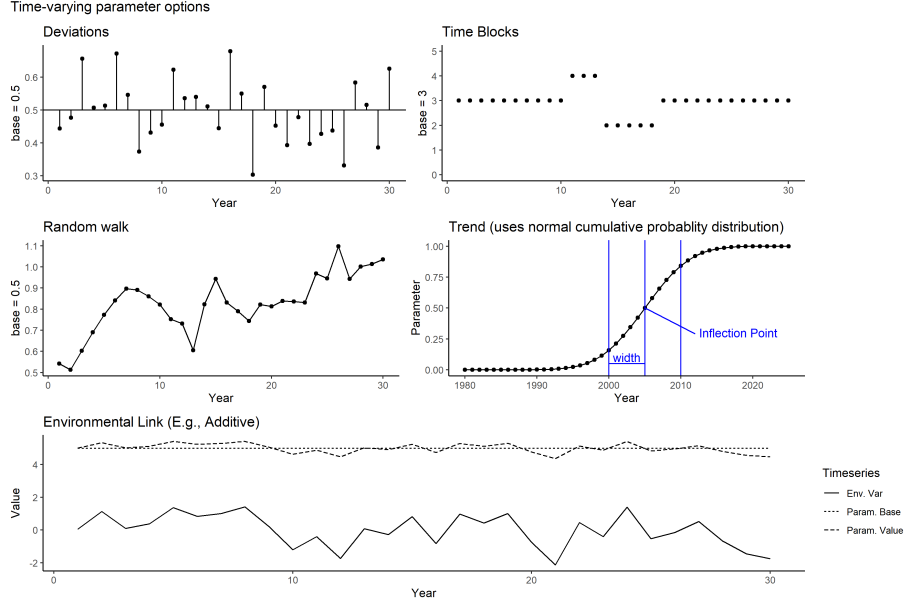
Figure 1: Some examples of time-varying setups.

option for the link function, while the tens and ones place is used to specify the environmental variable or derived quantity to which the parameter is linked. Note that environmental variables can only be included on an annual basis, so seasonal models would have the same effect applied to all seasons. If the environmental link and variable input is positive, then the parameter is linked to a variable specified in the data file environmental data; if it is negative, then the parameter is linked to a derived quantity. For example, env_var&link input 103 would use link type 1 and apply it to environmental data column 3, while the input -103 would use link type 1 and apply it to the "-3" column which is ln(relative summary biomass). The other options for both elements are enumerated below.

– The link function options (hundreds place) for the env_var&link input are:

* 1 = exponential scalar: $P_y = P_{base}e^{P_t E_y}$

* 2 = linear offset: $P_y = P_{base} + P_t E_y$

* 3 = Bounded replacement:
  $P_y = min(P_{base}) + \frac{max(P_{base}) - min(P_{base})}{1 + e^{P_t E_y + ln((P_{base} - min(P_{base}) + 0.0000001)/(max(P_{base}) - P_{base} + 0.0000001))}}$

* 4 = Logistic: $P_y = P_{base}\frac{2}{1 + e^{-P_{t2}(E_y - P_{t1})}}$

where:

207

* P_y = Parameter value in year y

* P_{base} = Base parameter value

* P_t = Link parameter value

* P_{t1} = First of 2 link parameters (offset)

* P_{t2} = Second of 2 link parameters (slope)

* E_y = Environmental index value or derived quantity value in year y

* $min(P_{base})$ = the minimum parameter bound of base parameter

* $max(P_{base})$ = the maximum parameter bound of base parameter

– The variable options (tens and ones place, or E_y) for the env_var&link input are either 1) a positive integer from 1 to 99 referencing a time-series located in the environmental data section of the data file, or 2) a negative value of -1 to -4 where E_y is one of the following model-derived quantities:

* -1; for ln(relative spawning biomass)

* -2; for recruitment deviation

* -3; for ln(relative summary biomass) (e.g., current year summary biomass divided by the unfished summary biomass)

* -4; for ln(relative summary numbers)

– The four derived quantities are all calculated at the beginning of each year within the model, so they are available to use as the basis for time-varying parameter links without violating any order of operations rules.

● Deviation Link (element 9). A positive integer invokes parameter deviations, but otherwise should be left as 0. SS3 expects the estimated deviations to be normal in distribution and the deviation values are multiplied by the standard error parameter as they are used. This differs from recruitment deviations and from the approach in SS3 v.3.24. Link options for parameter deviations are:

– 1 = multiplicative: $P_y = P_{base,y}e^{\text{dev}_y * \text{dev}_{se}}$,

– 2 = additive: $P_y = P_{base,y} + \text{dev}_y * \text{dev}_{se}$,

– 3 = random walk. Random walk options are implemented by using ρ in the objective function. $P_y = P_{base,y} + \sum_{n=1}^{y} \text{dev}_n * \text{dev}_{se}$

– 4 = mean reverting random walk with ρ.

  * $X_1 = \text{dev}_1 * \text{dev}_{se}$

  * $P_1 = P_{base,y} + X_1$

  * $X_y = \rho * X_{y-1} + \text{dev}_y * \text{dev}_{se}$

  * $P_y = P_{base,y} + X_y$

– 5 = mean reverting random walk with ρ and a logit transformation to stay within the minimum and maximum parameter bounds (approach added in v.3.30.16)

  * $X_1 = \text{dev}_1 * \text{dev}_{se}$

  * $R = P_{max} - P_{min}$

  * $Y_1 = ln(\frac{P_{base,y} - P_{min} + \text{nil}}{P_{max} - P_{base,y} + \text{nil}})$

  * $P_1 = P_{min} + \frac{R}{1 + e^{-Y_1 - X_1}}$. For the first year.

  * $X_y = \rho * X_{y-1} + \text{dev}_y * \text{dev}_{se}$

  * $Y_y = ln(\frac{P_{base,y} - P_{min} + \text{nil}}{P_{max} + P_{base,y} + \text{nil}})$

  * $P_1 = P_{min} + \frac{R}{1 + e^{-Y_y - X_y}}$. For years after the first year.

– 6 = mean reverting random walk with penalty to keep the root mean squared error (RMSE) near 1.0. Same as case 4, but with penalty applied.

– The option of extending the final model year deviation value subsequent years (i.e., into the forecast period) was added in v.3.30.13. This new option is specified by selecting the appropriate deviation link option and appending a 2 at the front (e.g, 25), which will use the final year deviation value for all forecast years.

where:

– P_y = Parameter value in year y

– $P_{base,y}$ = Base parameter value for year y

- dev_y = deviation in year y

- dev_{se} = standard error of the deviation

- nil is a small value (e.g., 0.0000001)

- Deviation Minimum Year (element 10). Year deviations start for the parameter. This must be specified if using parameter deviations, but otherwise should be left as 0.

- Deviation Maximum Year (element 11). Year deviations end for parameter. This must be specified if using parameter deviations, but otherwise should be left as 0.

- Deviation Phase (element 12). The phase in which the deviations for the parameter should be estimated. This must be specified if using parameter deviations, but otherwise should be left as 0.

- Use Time Blocks or Trends (element 13). Time blocks and trends are both specified using this input. If neither are used, this should be left as 0. For trend options, the cumulative normal distribution function is used as the shape of the trend in all cases, but the parameterization differs. In general, the trend used is:

$$P_y = P_{base} + P_{\text{offset}}\phi(\frac{y - P_{\text{infl}}}{P_{width}}) \tag{39}$$

where

- P_y is the final parameter value in year y

- P_{base} is the base parameter value

- P_{offset} is the parameter offset value

- ϕ is the standard cumulative normal distribution function

- P_{infl} is the inflection year (i.e., the year in which half of the total change from the base parameter has occurred)

- P_{width} is the standard deviation.

In all cases, 3 parameters are estimated and hence 3 short parameter lines are required. These parameter lines differ amongst the trend options.

The input value options for element 13 are:

- >0: time block index for parameter. See the time blocks section of the control file for more information on specifying time blocks.

– -1: Trend Offset option. Three parameters are estimated: end trend value as a
  logistic offset (input as $ln(P_{\text{offset}})$), inflection year logistic offset (input as
  $ln(P_{\text{infl}})$, and width (P_{width}. Offset trend value is in natural log space. Inflection
  year is also in natural log space and offset from $ln(0.5)$. Width is directly
  specified.

– -2: Trend Direct input option. In this case, $P_{\text{offset}} = 1$. Three parameters are
  input via short parameter lines: end trend parameter value (P_y where y is the
  final year), inflection year (P_{infl}, and width (P_{width}).

– -3: Trend Fractional option. In this case, $P_{\text{offset}} = 1$. Three parameters will be
  estimated: end trend parameter value as a fraction of base parameter maximum
  - minimum, inflection year as a fraction of end year - start year, and width
  (P_{width}). Width is directly input.

- Time Block Functional Form (element 14). Leave as 0, unless time blocks are used.

  – 0: multiplicative parameter ($P_{block} = P_{base} * e^{P_t}$)

  – 1: additive parameter ($P_{block} = P_{base} + P_t$)

  – 2: replace parameter ($P_{block} = P_t$)

  – 3: random walk across blocks ($P_{block} = P_{block,-1} + P_t$)

where:

  – P_{block} = Final parameter value in time block *block*

  – P_{base} = Base parameter value

  – P_t = Time-varying parameter value for a time block

  – $P_{block,-1}$ = Final parameter value in the previous time block

Code for the deviation link can be found in SS_timevaryparm.tpl, search for
"SS_Label_Info_14.3".

Specification of Time-Varying Parameters: Short Parameter Lines

If a time-varying specification set up in the long parameter lines for a particular section
requires additional parameters, short parameter lines need to be created following the long
parameter lines for the section (unless autogeneration is used, which creates short

parameter lines in control.ss_new upon running the model). The number of parameter lines required depends on the time-varying parameter specification.

For example, if two parameters were specified to have environmental linkages in the MG parameter section, below the MG parameters would be two parameter lines (when not auto-generating these lines), which is an environmental linkage parameter for each time-varying base parameter:

LO	HI	INIT	Prior Value	Prior SD	Prior Type	Phase	Parameter Label
COND: Only if MG parameters are time-varying							
-99	99	1	0	0.01	0	-1	#Wtlen_1_Fem_ENV_add
-99	99	1	0	0.01	0	-1	#Wtlen_2_Fem_ENV_add

In Stock Synthesis v.3.30, the time-varying input short parameter lines are organized such that all parameters that affect a base parameter are clustered together with time blocks (or trend) first, then environmental linkages, then parameter deviations. For example, if the mortality-growth (MG) base parameters 3 and 7 had time varying changes, the order would look like:

MG base parameter 3	Block parameter 3-1
	Block parameter 3-2
	Environmental link parameter 3-1
	Deviation se parameter 3
	Deviation ρ parameter 3
MG base parameter 7	Block parameter 7-1
	Deviation se parameter 7
	Deviation ρ parameter 7

The number of short parameter lines for each time-varying setup selected depends on the selection options. The autogeneration feature can be used to figure out which parameter lines are needed. The short parameter lines needed for different time-varying options are:

- Environmental Linkages: Requires 1 short parameter line (P_t), except for link option 4, which requires 2 short parameter lines (P_{t1} and P_{t2}).

212

- Parameter deviations: Requires 2 short parameter lines, one for the standard error (dev_{se}), followed by one for ρ. Note that an input for ρ is required but only used with random walk options. For the random walk options, ρ can be set at 1 for a random walk with no drift or >1 for a random walk with drift.

- Time Blocks: One parameter for each time block (P_t) set up in the pattern.

- Trends: Requires 3 parameter lines. The interpretation of the parameters differs by the trend option selected, but in general they are a parameter for the final parameter value, a parameter for the inflection point year, and a parameter for the width (i.e., the standard deviation).

Example Time-varying Parameter Setups

The time-varying parameter options in Stock Synthesis are flexible. Below are some example setups that illustrate how the time-varying options could be used in a model, although there are many more possible setups.

Environmental and density dependent linkages

- Suppose growth rate is found to be linked with an index of water temperature. The water temperature proxy could be input into the data file as environmental data. If it is input as index number 1, the growth parameter K (if using a von Bertalanffy growth equation) could be linked to the water temperature proxy data by specifying the code "201" in the env_var&link function input. This would establish an offset link between the parameter and the temperature proxy. One additional parameter line is required after the "MG parameter" long parameter lines section.

- Suppose for a fishery, selectivity is thought to shift depending on population size. Smaller fish are selected when there are lower population numbers, while larger fish are selected when there are higher population numbers. The selectivity parameter could be made time-varying using the code "-104" in the env_var&link option, which assumes a exponential scalar link between the base selectivity parameter and the time varying parameter value. One additional parameter line is required at the end of the selectivity long parameter lines section.

Parameter Deviations

- Suppose a selectivity parameter is thought to drift every year during 2000-2010. This could be represented using a random walk link option available within the parameter deviations options. To implement this, the user could input 3 into the "dev link" input on the long parameter line for the selectivity parameter, and then input values 2000 and 2010 for "dev min yr" and "dev max yr", respectively. The dev phase could

be set to 3. With this setup, 2 additional short parameter lines would be expected, one for the standard error and one for ρ. Both of these will be used since a random walk option is selected. To use a random walk without drift, ρ is set at 1 with a negative phase.

Time Blocks

- Offset approach: One or more time blocks are created and cover all or a subset of the years. Each block gets a parameter that is used as an offset from the base parameter (time block functional form 1). In this situation, typically the base parameter and each of the offset parameters are estimated. In years not covered by blocks, the base parameter alone is used. However, if blocks cover all the years, then the value of the block parameter is completely correlated with the mean of the block offsets, so model convergence and variance estimation could be affected. The recommended approach when using offsets is to not have all years covered by blocks or to fix the base parameter value at a reasonable level when doing offsets for all years.

- Replacement approach, Option A: Time blocks are created which cover a subset of the years. The base parameter is used in the non-block years and the value of the base parameter is replaced by the block parameter in each respective block (time block functional form 2). In this situation, typically the base parameter and each of the block parameters are estimated.

- Replacement approach, Option B: Replacement time blocks are created for all the years, so the base parameter is simply a placeholder that is always replaced by a block parameter (time block functional form 2). In this situation, do not allow the model to estimate the base parameter and only estimate the corresponding block replacement parameters, otherwise, the search algorithm will be attempting to estimate parameters that do not contribute to the log likelihood, so model convergence and variance estimation could be affected.

Trends

- Suppose natural mortality was thought to increase from 0.1 to 0.2 during 2000 to 2010. This could be input as a trend. First, the natural mortality parameter would be fixed at an initial value of 0.1. Then, a value of -2 could be input into the "use block" column of the natural mortality long parameter line to indicate that the direct input option for trends should be used. The long parameter line for M could look like:

LO	HI	INIT	<other entries>	PHASE	<other entries>	Use_Block	Block Fxn	Parameter Label
0	4	0.1	...	-1	...	-2	0	#M

- Three short parameter lines are then expected after the mortality-growth long parameter lines, one for the final value, one for the inflection year and one for the width. The final value could be fixed by using 0.2 as the final value on the short parameter line and a negative phase value. The inflection year could be fixed at 2005 by inputting 2005 for the inflection year in the short parameter line with a negative phase. Finally, the width value (i.e., standard deviation of the cumulative normal distribution) could be set at 3 years. The short parameter lines could look like:

LO	HI	INIT	Prior Value	Prior SD	Prior Type	Phase	Parameter Label
0.001	4	0.2	0	0.01	0	-1	#M_TrendFinal
1999	2011	2005	0	0.01	0	-1	#M_TrendInfl
-99	99	3	0	0.01	0	-1	#M_TrendWidth_yrs

Time-Varying Growth Considerations

When time-varying growth is used, there are some additional considerations to be aware of:

- Growth in the forecast with time blocks: Growth deviations propagate into the forecast because growth is by cohort according to the current year's growth parameters. The user can select which growth parameters get used during the forecast by setting the end year of the last block, if using time blocks. If the last block ends in the model's end year, then the growth parameters in effect during the forecast will be the base parameters. By setting the end year of the last block to one year past the model end year (endyr), the model will continue the last block's growth parameter levels throughout the forecast.

- The equilibrium benchmark quantities (MSY, F40%, etc.) previously used the model end year's (endyr) body size-at-age, which is not in equilibrium. Through the forecast file, it is possible to specify a range of years over which to average the size-at-age used in the benchmark calculations. An option to create equilibrium growth from averaged growth parameters would be a more realistic option and is under consideration, but is not yet available.

Time-Varying Stock-Recruitment Considerations

- The σ_R and autocorrelation parameters cannot be time-varying.

- The autocorrelation parameter cannot be estimated accurately within SS3 (Johnson et al., 2016), so external (i.e., external to SS3) estimation for selecting an autocorrelation value is currently recommended. The autocorrelation of the recruitment deviations appears in the report file, which can aid in selecting the autocorrelation value.

- The value of R0 and steepness in the initial year are used within virgin calculations and within the benchmarks for calculation of the denominator in depletion estimates. The average value of R0 and steepness in the range of years specified as the benchmark years inputs 9 and 10 (see the forecast file specifications) is used for MSY-type calculations.

- The spawner-recruit regime parameter is a modifier on R0. The regime shift parameter line allows for multi-year or environmentally driven deviations from R0 without changing R0 itself. The regime shift base parameter should have a base value of 0.0 and not be estimated (i.e., have a negative phase). Similar to the cohort-growth deviation, it serves simply as a base for adding time-varying adjustments.

- The same algebraic effect on the calculated recruitment can be achieved by different combinations of spawner-recruit parameter options (e.g., changing R0 directly instead of the regime shift parameter). It is recommended to use block, trend or environmental effects on R0 only for long-term effects, and use time-vary effects on the regime shift parameter for transitory but multi-year deviations from R0.

- If the R0 or steepness parameters are time-varying, then the model will use the current year's parameters to calculate the expected value of recruits as a function of the spawning biomass, then applies the recruitment deviations. If the regime shift parameter is time-varying, then the model applies the change in the regime shift parameter after calculating the expected value of recruits as a function of spawning biomass.

Forecast Considerations with Time-Varying Parameters

Users should judiciously consider which parameter values are applied during forecast years. SS3 will default to use all base parameter values during the forecast period, but alternatively, which years of selectivity, relative F, and recruitment should be used during the forecast period by specifying in the forecast file.

Time-varying parameters can extend into the forecast period. For example, a parameter with a time block that stops at the model end year will revert to the base parameter value

216

for the forecast, but when the block definition extends to include some or all forecast years, the last block will apply to the forecast. A good practice is to use 9999 as the terminal year for the last block to ensure including all forecast years. If a parameter has deviations and the deviations' year range includes the forecast years, then the parameter will have process uncertainty in the forecast years and MCMC draws(if using) will include the variability.

Parameterizing the Two-Dimensional Autoregressive Selectivity

When the two-dimensional autoregressive selectivity feature is turned on for a fleet, the selectivity is calculated as a product of the assumed selectivity pattern and a non-parametric deviation term deviating from this assumed pattern:

$$\hat{S}_{a,t} = S_a exp^{\epsilon_{a,t}} \tag{40}$$

where S_a is specified in the corresponding age/length selectivity types section and it can be either parametric (recommended) or non-parametric (including any of the existing selectivity options in SS3); $\epsilon_{a,t}$ is simulated as a two-dimensional first-order autoregressive (2D AR1) process:

$$vec(\epsilon) \sim MVN(\mathbf{0}, \sigma_s^2 \mathbf{R_{total}}) \tag{41}$$

where ϵ is the two-dimensional deviation matrix and $\sigma_s^2 \mathbf{R_{total}}$ is the covariance matrix for the 2D AR1 process. More specifically, σ_s^2 quantifies the variance in selectivity deviations and $\mathbf{R_{total}}$ is equal to the kronecker product ($\otimes$) of the two correlation matrices for the among-age and among-year AR1 processes:

$$\mathbf{R_{total}} = \mathbf{R} \otimes \tilde{\mathbf{R}} \tag{42}$$

$$\mathbf{R}_{a,\tilde{a}} = \rho_a^{|a-\tilde{a}|} \tag{43}$$

$$\tilde{\mathbf{R}}_{t,\tilde{t}} = \rho_t^{|t-\tilde{t}|} \tag{44}$$

where ρ_a and ρ_t are the among age and among year AR1 coefficients, respectively. When both of them are zero, $\mathbf{R}$ and $\tilde{\mathbf{R}}$ are two identity matrices and their Kronecker product,

$\mathbf{R_{total}}$, is also an identity matrix. In this case selectivity deviations are essentially identical and mutually independent:

$$\epsilon_{a,t} \sim N(0, \sigma_s^2) \tag{45}$$

Using the Two-Dimensional Autoregressive Selectivity

Note, Xu et al. (2019) has additional information on tuning the 2D AR selectivity parameters. First, fix the two AR1 coefficients (ρ_a and ρ_t) at 0 and tune σ_s iteratively to match the relationship:

$$\sigma_s^2 = SD(\epsilon)^2 + \frac{1}{(a_{max} - a_{min} + 1)(t_{max} - t_{min} + 1)} \sum_{a=a_{min}}^{a_{max}} \sum_{t=t_{min}}^{t_{max}} SE(\epsilon_{a,t})^2 \tag{46}$$

The minimal and maximal ages/lengths and years for the 2D AR1 process can be freely specified by users in the control file. However, we recommend specifying the minimal and maximal ages and years to cover the relatively "data-rich" age/length and year ranges only. Particularly we introduce:

$$b = 1 - \frac{\frac{1}{(a_{max} - a_{min} + 1)(t_{max} - tmin + 1)} \sum_{a=a_{min}}^{a_{max}} \sum_{t=t_{min}}^{t_{max}} SE(\epsilon_{a,t})^2}{\sigma_s^2} \tag{47}$$

as a measure of how rich the composition data is regarding estimating selectivity deviations. We also recommend using the Dirichlet-Multinomial method to "weight" the corresponding composition data while σ_s is interactively tuned in this step.

Second, fix σ_s at the value iteratively tuned in the previous step and estimate $\epsilon_{a,t}$. Plot both Pearson residuals and $\epsilon_{a,t}$ out on the age-year surface to check their 2D dimensions. If their distributions seems to be not random but rather be autocorrelated (deviation estimates have the same sign several ages and/or years in a row), users should consider estimating and then including the autocorrelations in $\epsilon_{a,t}$.

Third, extract the estimated selectivity deviation samples from the previous step for estimating ρ_a and ρ_t externally by fitting the samples to a stand-alone model written in Template-Model Builder (TMB). In this model, both ρ_a and ρ_t are bounded between 0 and 1 via applying a logic transformation. If at least one of the two AR1 coefficients are notably different from 0, the model should be run one more time by fixing the two AR1 coefficients at their values externally estimated from deviation samples. The Pearson residuals and $\epsilon_{a,t}$ from this run are expected to distribute more randomly as the autocorrelations in selectivity deviations can be at least partially included in the 2D AR1 process.

Continuous seasonal recruitment

Setting up a seasonal model such that recruitment can occur with similar and independent probability in any season of any year is awkward in SS3. Instead, SS3 can be set up so that each quarter appears as a year (i.e., a seasons as years model). All the data and parameters are set up to treat quarters as if they were years. Note that setting up a seasons as years model also requires that all rate parameters be re-scaled to correctly account for the quarters being treated as years.

Other adjustments to make when using seasons as years include:

- Re-index all "year seas" inputs to be in terms of quarter-year because all are now season 1; increase end year (endyr) value in sync with this.

- Increase max age because age is now in quarters.

- In the age error definitions, increase the number of entries to reflect that age is now in quarters.

- In the age error definitions, recode so that each quarter-age gets assigned to the correct age bin. This is because the age data are still in terms of age bins; i.e., the first 4 entries for quarter-ages 1 through 4 will all be assigned to age bin 1.5; the next four to age bin 2.5; you cannot accomplish the same result by editing the age bin values because the standard deviation of ageing error is in terms of age bin.

- In the control file, multiply the natural mortality age breakpoints and growth Amin and Amax values by 1/season duration.

- Decrease the R0 parameter starting value because it is now the average number of recruitments per quarter year.

- Edit the recruitment deviation (rec_dev) start and end years to be in terms of quarter year.

- Edit any age selectivity parameters that refer to age, because they are now in terms of quarter age.

- If there needs to be some degree of seasonality to a parameter, then you could create a cyclic pattern in the environmental input and make the parameter a function of this cyclic pattern.

Detailed Information on Stock Synthesis Processes

The processes and calculations within SS3 can be complex and not transparent based on the model input files. Here, additional information on processes within SS3 is provided.

Jitter

The following steps are now performed to determine the jittered starting parameter values (illustrated in Figure 2):

1. A normal distribution is calculated such that the $pr(P_{MIN}) = 0.1\%$ and the $pr(P_{MAX}) = 99.9\%$.

2. A jitter shift value, termed "K", is calculated from the distribution equal to $pr(P_{CURRENT})$.

3. A random value is drawn, "J", from the range of K-jitter to K+jitter.

4. Any value which falls outside the 0-1 range (in the cumulative normal space) is mapped back from the bound to a point one-tenth of the way from the bound to the initial value.

5. J is a new cumulative normal probability value.

6. Calculate a new parameter value, $P_{JITTERED}$, such that $pr(P_{JITTERED}) = J$.

In SS3, the jitter fraction defines a uniform distribution in cumulative normal space +/- the jitter fraction from the initial value (in cumulative normal space). The normal distribution for each parameter, for this purpose, is defined such that the minimum bound is at 0.001, and the maximum at 0.999 of the cumulative distribution. If the jitter faction and original initial value are such that a portion of the uniform distribution goes beyond 0.001 or 0.999 of the cumulative normal, the new value is set to one-tenth of the way from the bound to the original initial value.

Therefore sigma = (max-min) / 6.18. For parameters that are on the log-scale, sigma may be the correct measure of variation for jitters, for real-space parameters, CV (= sigma/original initial value) may be a better measure.

If the original initial value is at or near the middle of the min-max range, then for each 0.1 of jitter, the range of jitters extends about 0.25 sigmas to either side of the original value (though as the total jitter increases the relationship varies more than this), and the average
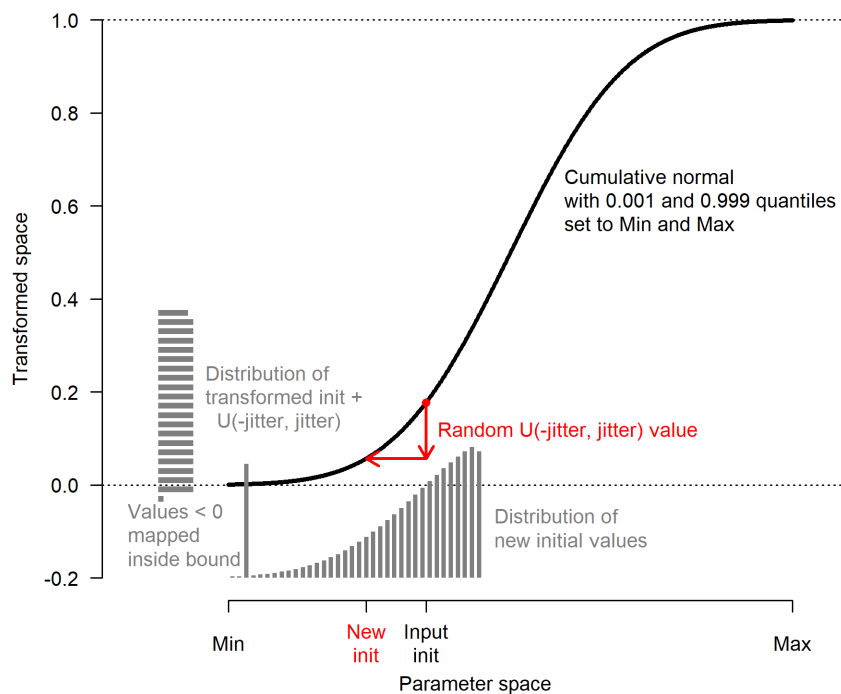
Figure 2: Illustration of the jitter algorithm

absolute jitter is about half of that. For values far from the middle of the min-max range, the resulting jitter is skewed in parameter space, and may hit the bound, invoking the resetting mentioned above.

To evaluate the jittering, the bounds, and the original initial values, a jitter_info table is available from r4ss, including sigma, CV and InitLocation columns (the latter referring to location within the cumulative normal - too close to 0 or 1 indicates a potential issue).

Note: parameters with min $\leq$ -99 or max $\geq$ 999 are not jittered to avoid unreasonable values (a warning is produced to indicate this).

Parameter Priors

Priors on parameters fulfill two roles in SS3. First, for parameters provided with an informative prior, SS3 is receiving additional information about the true value of the parameter. This information works with the information in the data through the overall log likelihood function to arrive at the final parameter estimate. Second, diffuse priors provide only weak information about the value of a prior and serve to manage model performance during execution. For example, some selectivity parameters may become unimportant depending upon the values of other parameters of that selectivity function. In the double normal selectivity function, the parameters controlling the width of the peak

and the slope of the descending side become redundant if the parameter controlling the final selectivity moves to a value indicating asymptotic selectivity. The width and slope parameters would no longer have any effect on the log likelihood, so they would have no gradient in the log likelihood and would drift aimlessly. A diffuse prior would then steer them towards a central value and avoid them crashing into the bounds. Another benefit of diffuse priors is the control of parameters that are given unnaturally wide bounds. When a parameter is given too broad of a bound, then early in a model run it could drift into this tail and potentially get into a situation where the gradient with respect that parameter approaches zero even though it is not at its global best value. Here the diffuse prior helps move the parameter back towards the middle of its range where it presumably will be more influential and estimable.

The options for parameter priors are described as a function of $Pval$, the value of the parameter for which a prior is being calculated, as well as the parameter bounds in the case of the beta distribution ($Pmax$ and $Pmin$), and the input values for $Prior$ and Pr_SD, which in some cases are the mean and standard deviation, but interpretation depends on the prior type. The Prior Likelihoods below represent the negative log likelihood in all cases.

Prior Types

Note that the numbering in v.3.30 is different from that used in v.3.24 (where confusingly -1 indicated no prior and 0 indicated a normal prior). The calculation of the negative log likelihood is provided below for each prior types, as a function of the following inputs:

P_{init} The value of the parameter for which a prior is being calculated where init can either be the initial un-estimated value or the estimated value (3rd column in control or control.ss_new file)

P_{LB} The lower bound of the parameter (1st column in control file)

P_{UB} The upper bound of the parameter (2nd column in control file)

P_{PR} The input value for the prior input (4th column in control file)

P_{PRSD} The standard deviation input value for the prior (5th column in control file)

- Prior Type $= 0 =$ No prior applied
  In a Bayesian context this is equivalent to a uniform prior between the parameter bounds.

- Prior Type $= 1 =$ Symmetric beta prior
  The symmetric beta is scaled between parameter bounds, imposing a larger penalty near the bounds. Prior standard deviation of 0.05 is very diffuse and a value of 5.0 provides a smooth U-shaped prior. The prior input is ignored for this prior type.

$$\mu = -P_{\text{PRSD}} \cdot ln\left(\frac{P_{\text{UB}} + P_{\text{LB}}}{2} - P_{\text{LB}}\right) - P_{\text{PRSD}} \cdot ln(0.5) \tag{48}$$

$$\text{Prior Likelihood} = -\mu - P_{\text{PRSD}} \cdot ln\left(P_{\text{init}} - P_{\text{LB}} + 0.0001\right) -$$
$$P_{\text{PRSD}} \cdot ln\left(1 - \frac{P_{\text{init}} - P_{\text{LB}} - 0.0001}{P_{\text{UB}} - P_{\text{LB}}}\right) \tag{49}$$
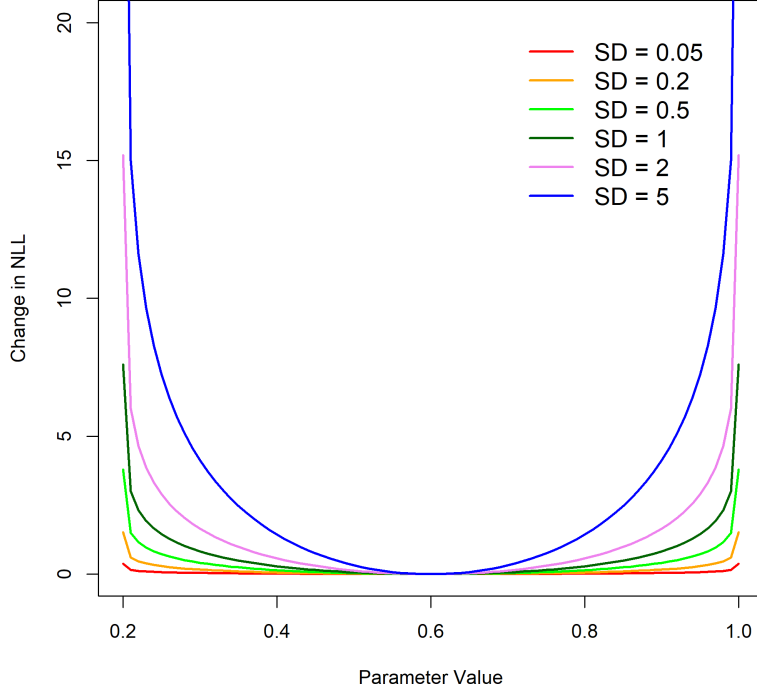

Figure 3: The shape of the symmetric beta prior across alternative standard deviation values and the change in the negative log-likelihood.

- Prior Type = 2 = Beta prior
  The definition of μ is consistent with CASAL's formulation with the β_{PR} and α_{PR} corresponding to the m and n parameters.

$$\mu = \frac{P_{\text{PR}} - P_{\text{LB}}}{P_{\text{UB}} - P_{\text{LB}}} \tag{50}$$

$$\tau = \frac{(P_{\text{PR}} - P_{\text{LB}})(P_{\text{UB}} - P_{\text{PR}})}{P_{\text{PRSD}}^2} - 1 \tag{51}$$

$$\beta_{\text{PR}} = \tau \cdot \mu \tag{52}$$

$$\alpha_{\text{PR}} = \tau(1 - \mu) \tag{53}$$

$$\begin{aligned}
\text{Prior Likelihood} = &(1 - \beta_{\text{PR}}) \cdot ln(0.0001 + P_{\text{init}} - P_{\text{LB}}) + \\
&(1 - \alpha_{\text{PR}}) \cdot ln(0.0001 + P_{\text{UB}} - P_{\text{init}}) - \\
&(1 - \beta_{\text{PR}}) \cdot ln(0.0001 + P_{\text{PR}} - P_{\text{LB}}) - \\
&(1 - \alpha_{\text{PR}}) \cdot ln(0.0001 + P_{\text{UB}} - P_{\text{PR}})
\end{aligned} \tag{54}$$

- Prior Type 3 = Lognormal prior
  Note that this is undefined for $p <= 0$ so the lower bound on the parameter must be > 0. The prior value is input into the parameter line in natural log space while the initial parameter value is defined in normal space (e.g., init = 0.20, prior = -1.609438).

$$\text{Prior Likelihood} = \frac{1}{2}\left(\frac{ln(P_{\text{init}}) - P_{\text{PR}}}{P_{\text{PRSD}}}\right)^2 \tag{55}$$

- Prior Type 4 = Lognormal prior with bias correction
  This option allows the prior mean value to be entered as the ln(mean). Note that this is undefined for $p <= 0$ so the lower bound on the parameter must be > 0.

$$\text{Prior Likelihood} = \frac{1}{2}\left(\frac{ln(P_{\text{init}}) - P_{\text{PR}} + \frac{1}{2}P_{\text{PRSD}}^2}{P_{\text{PRSD}}}\right)^2 \tag{56}$$

- Prior Type 5 = Gamma prior
  The lower bound should be 0 or greater.

$$\text{scale} = \frac{P_{\text{PRSD}}^2}{P_{\text{PR}}} \tag{57}$$

$$\text{shape} = \frac{P_{\text{PR}}}{\text{scale}} \tag{58}$$

$$\text{Prior Likelihood} = -\text{shape} \cdot ln(\text{scale}) - ln\big(\Gamma(\text{shape})\big) + (\text{shape} - 1) \cdot ln(P_{\text{init}}) - \frac{P_{\text{init}}}{\text{scale}} \tag{59}$$

- Prior Type 6 = Normal prior
  Note that this function is independent of the parameter bounds.

$$\text{Prior Likelihood} = \frac{1}{2}\left(\frac{P_{\text{init}} - P_{\text{PR}}}{P_{\text{PRSD}}}\right)^2 \tag{60}$$

Forecast Module: Benchmark and Forecasting Calculations

Stock Synthesis v.3.20 introduced substantial upgrades to the benchmark and forecast module. The general intent was to make the forecast outputs more consistent with the requirement to set catch limits that have a known probability of exceeding the overfishing limit. In addition, this upgrade addressed several inadequacies with the previous module, including:

- The average selectivity and relative F was the same for the benchmark and the forecast calculations;

- The biology-at-age in endyr+1 was used as the biology for the benchmark, but biology-at-age propagated forward in the forecast if there was time-varying growth;

- The forecast module had a inefficient approach to calculation of overfishing limit (OFL) conditioned on previously catching ABC;

- The forecast module implementation of catch caps was incomplete and applied some caps on a seasonally, rather than the more logical annual basis;

- The Fmult scalar for fishing intensity presented a confusing concept for many users;

- No provision for specification of catch allocation among fleets;

- The forecast allowed for a blend of fixed input catches and catches calculated from target F; this is not optimal for calculation of the variance of F conditioned on a catch policy that sets annual catch limits (ACLs).

The v.3.20 module addressed these issues by:

- Providing for unique specification of a range of years from which to calculate average selectivity for benchmark, average selectivity for forecast, relative F for benchmark, and relative F for forecast;

- Create a new specification for the range of years over which to average size-at-age and fecundity-at-age for the benchmark calculation. In a setup with time-varying growth, it may make sense to do this over the entire range of years in the time series. Note that some additional quantities still use their endyr values, notably the migration rates and the allocation of recruitments among areas. This will be addressed shortly;

- Create a multiple pass approach that rectifies the OFL calculation;

- Improve the specification of catch caps and implement specification of catch allocations so that there can be an annual cap for each fleet, an annual cap for each

area, and an annual allocation among groups of fleets (e.g., all recreational fleets vs. all commercial fleets);

- Introduce capability to have implementation error in the forecast catch (single value applied to all fleets in all seasons of the year).

Multiple Pass Forecast

The most complicated aspect of the changes is with regard to the multiple pass aspect of the forecast. This multiple pass approach is needed to calculate both OFL and ABC in a single model run. More importantly, the multiple passes are needed in order to mimic the actual sequence of assessment-management action - catch over a multi-year period. The first pass calculates OFL based on catching OFL each year, so presents the absolute maximum upper limit to catches. The second pass forecasts a catch based on a harvest policy, then applies catch caps and allocations, then updates the F's to match these catches. In the third pass, stochastic recruitment and catch implementation error are implemented and SS3 calculates the F that would be needed in order to catch the adjusted catch amount previously calculated in the second pass. With this approach, SS3 is able to produce improved estimates of the probability that F would exceed the overfishing F. In effect it is the complement of the P* approach. Rather than the P* approach that calculates the stream of annual catches that would have an annual probability of F>Flimit, SS3 calculates the expected time series of P* that would result from a specified harvest policy implemented as a buffer between Ftarget and Flimit.

The sequence of multiple forecast passes is as follows:

1. Pass 1 (a.k.a. Fcast_Loop1)

    (a) Loop Years

        i. SubLoop (a.k.a. ABC_Loop) = 1

            A. R = f(SSB) with no deviations

            B. F = Flimit

            C. Fixed input catch amounts ignored

            D. No catch adjustments (caps and allocations)

            E. No implementation error

            F. Result: OFL conditioned on catching OFL each year

2. Pass 2

(a) Loop Years

    i. SubLoop = 1

        A. R = f(SSB) with no deviations

        B. F = Flimit

        C. Fixed input catch amounts ignored

        D. No catch adjustments (caps and allocations)

        E. No implementation error

        F. Result: OFL conditioned on catching ABC previous year. Stored in std_vector.

    ii. SubLoop = 2

        A. R = f(SSB) with no deviations

        B. F = Ftarget to get catch for each fleet in each season

        C. Fixed input catch amounts replace catch from step 2

        D. Catch adjustments (caps and allocations) applied on annual basis (after looping through seasons and and areas within this year). These adjustments utilize the logistic joiner approach so the overall results remain completely differentiable.

        E. No implementation error

        F. Result: ABC as adjusted for caps and allocations

    iii. SubLoop = 3

        A. R = f(SSB) with no deviations

        B. Catches from Pass 2 multiplied by the random term for implementation error

        C. F = adjusted to match the catch*error while taking into account the random recruitments. This is most easily visualized in a MCMC context where the recruitment deviation and the implementation error deviations take on non-zero values in each instance. In MLE, because the forecast recruitments and implementation error are estimated

parameters with variance, their variance still propagates to the derived quantities in the forecast.

D. Result: Values for F, SSB, Recruitment, Catch are stored in std-vectors

- In addition, the ratios F/Flimit and SSB/SSBlimit or SSB/SSBtarget are also stored in std_vectors.

- Estimated variance in these ratios allows calculation of annual probability that F > Flimit or B < Blimit. This is essentially the realized P* conditioned on the specified harvest policy.

Example Effects on Correlations

An example that illustrates the above process was conducted. The situation was a low M, late-maturing species, so changes are not dramatic. The example conducted a 10 year forecast and examined correlations with derived quantities in the last year of the forecast. This was done once with the full set of 3 passes as described above, and again with only 2 passes and stochastic recruitment occurring in pass 2, rather than 3. This alternative setup is more similar to forecasts done using previous model versions.

	2 Forecast Passes with F from ABC and random recruitment				2 Forecast Passes with catch from target F and equilibrium recruitment		
	Factor X	Factor Y	Corr		Factor X	Factor Y	Corr
A1	F 2011	RecrDev 2002	-0.126	A2	F 2011	RecrDev 2002	0.090
B1	F 2011	Recr 2002	0.312	B2	F 2011	Recr 2002	0.518
C1	ForeCatch 2011	RecrDev 2002	0.000	C2	ForeCatch 2011	RecrDev 2002	0.129
D1	ForeCatch 2011	Recr 2002	0.455	D2	Forecatch 2011	Recr 2002	0.555

Correlation A2 shows a small positive correlation between the recruitment deviation in 2002 and the F in 2011. This is probably due to the fact that a positive deviation in recruitment in 2002 will reduce the chances that the biomass in 2011 will be below the inflection point in the control rule. This occurs because in calculating catch from F, the model effectively "knows" the future recruitments. I predict that this B1 correlation would be near zero if there was no inflection in the control rule.

Correlation A1 shows this turning into a negative correlation. This is because the future catches are first calculated from equilibrium recruitment, then when random recruitments

are implemented, a positive recruitment deviation will cause a negative deviation in the F needed to catch that now "fixed" amount of future catch.

Correlations B1 and B2 are in terms of absolute recruitment, not recruitment deviation. Now overall model conditions that cause a higher absolute recruitment level will also result in a higher forecast level. No surprise there, and the correlation is stronger when variance is based on catch is calculated from F (B2).

Correlation C2 shows a positive correlation between recruitment deviation in 2002 and forecast catch in 2011. However, correlation C1 is 0.0 because the forecast catch in 2011 is set based on equilibrium recruitment and is not influenced by the recruitment deviations.

Future Work

- More testing with high M, rapid turnover conditions

- Testing without inflection in control rule

- Consider separating implementation error into a pass #4 so results will more clearly show effect of assessment uncertainty separate from implementation uncertainty

- Consider adding a random "assessment" error which essentially is a random variable that scales population abundance before passing into the forecast stage. Complication is figuring out how to link it to the correlated error in the benchmark quantities

- Because all of these calculations occur only in the standard deviation phase (sdphase) or the MCMC evaluation (mceval) phase, it would be feasible for mceval calls to add an additional pass that is implemented many times and in which random forecast recruitment draws are made.

- Factors like selectivity and fleet relative F levels are calculated as an average of these values during the time series. This is internally consistent if these factors do not vary during the time series (although clearly this is a stiff model that will underestimate process variance. However, if these factors do vary over time, then the average used for the forecast will under-represent the variance. A better approach would be to set up the parameters of selectivity as a random process that extends throughout the forecast period, and to update estimated selectivity in each year of the forecast based upon the random realization of these parameters.

Fishing Mortality in Stock Synthesis

The implementation and reporting of fishing mortality rate, F, in SS3 has some aspects that can be confusing. This description provides an overview of the ways in which F is calculated, used, and reported.

Rationale

Fishery management systems expect to have a measure of annual fishing mortality that describes the intensity of the fishery such that an optimal level of F can be articulated and accountability measures can be invoked if F is too high, e.g., overfishing. This concept is simple and straightforward if the model is a simple biomass dynamics such that a single annual F value operates on the entirety of a non-age structured population. It also is simple for age-structured models that have a single fishing fleet and knife-edge selectivity beginning at some specified age.

The simplicity of F disappears quickly as models invoke a variety of realistic complexities such as: allowing the F to differ among ages or to be based on size; using a collection of fleets with different F levels and different age patterns for F; spreading the population across areas and allowing different fleets with different F among the areas. An unambiguous measure of annual fishing intensity that represents the cumulative effect of all that complexity has not been defined. This problem has not been solved with SS3, but some logical alternatives have been made available.

Nomenclature

The nomenclature below ignores sex, morphs and areas for simplicity. The quantities associated with F calculations are defined as:

f is fleet.

t is a time step; continuous across years y and seasons s; equivalent to year if only 1 season.

a is age.

$C_{t,f}$ is fleet-specific catch in a time step.

$B_{t,f}$ is fleet specific available biomass, e.g., total biomass filtered by fleet-specific age selectivity, $s_{t,f,a}$.

$s_{t,f,a}$ is age-specific selectivity for a fleet. If selectivity is length-specific, then age-specific selectivity is calculated as the dot product across length bins of length selectivity and the normal (or lognormal) distribution of length-at-age. If selectivity is both length- and age-based, which is an entirely normal concept in SS3, then age selectivity due to length selectivity is calculated first, then multiplied by the direct age selectivity. This compound

age selectivity is used in the mortality calculations and is reported as asel2 in report file. See appendix to Methot and Wetzel (2013) for more detail on this.

$F'_{t,f}$ is the apical fishing mortality for a fleet. This means that it is the rate for the age that has selectivity equal to 1.0. If your model is using F's as parameters, then the parameter values are for F'.

$F_{t,f,a}$ is age and fleet-specific fishing mortality rate equal to $F'_{t,f} * s_{t,f,a}$. Note that it is possible for no age to have a selectivity equal to 1.0. In this case, F' is still the rate for the hypothetical age that has selectivity equal to 1.0. The reported F' values are not rescaled to be an F for the age with peak selectivity. Users need to take this into account if they are comparing reported F' values to reported vector of $F_{t,f,a}$ values.

annF_y is a measure of the total fishing intensity for a year, based on one of several user-specified options (see below).

Fstd$_y$ is a standardized measure of the total fishing intensity for a year and is reported in the derived quantities, so variance is calculated for this quantity. See below for how it relates to $annF$.

Terminology and reporting of ann$F$ and Fstd has been slightly revised for clarity in v.3.30.15.00 and the description here follows the new conventions.

F Calculation
SS3 allows for three approaches to estimate the F' that will match the input values for retained catch. Note that SS3 is calculating the F' to match the retained catch conditional on the fraction of total catch that is retained, e.g., the total catch is partitioned into retained and discarded portions.

1. Pope's method decays the numbers-at-age to the middle of the season, calculates a harvest rate for each fleet, $H_{t,f}$, that is the ratio of $C_{t,f}$ to $B_{t,f}$, then decays the survivors to the end of the season. The total mortality, $Z_{t,a}$, from the ratio of survivors to initial numbers, is then calculated. The Z is subsequently used for in-season interpolation to get expected values for observations.

2. F as parameters uses the standard Baranov catch equation and lets ADMB find the F' parameter values that produce the lowest negative log-likelihood, which includes fit to the input catch data. F as parameters method tends to work better than Pope's or hybrid in high F situations because it allows for some lack of fit to catch levels in early iterations and can later improve this fit as it closes in on the best solution.

3. Hybrid F starts by calculating a harvest rate, H, using Pope's, then converts these H values, which have units of fractional harvest rate, into an approximate of F' in

exponential units, tuning these F' values over a few iterations to get a better match to each fleet's catch.

Items to note:

- SS3 includes a permutation on the F as parameters method. In the first few phases, SS3 uses hybrid, then between phases it converts these directly calculated F' values into parameters and proceeds in subsequent phases and MCMC to use the parameter approach. This variation on the parameter method is the recommend approach in high F situations.

- With Pope's method, the H values are fraction caught, so duration of the season does not matter. Parameter and hybrid treat F' identically and multiply the F' values by season duration (which has units of fraction of a year) as it is used. Each of the F methods ends up with a $Z_{t,f}$ that is used for in-season interpolation.

Relative F and Fmult

The F' is fleet-specific, so it is useful to have a concept of relative F, relF_f, among fleets. In SS3, rel$F_f = F'_{t,f}/\sum_f F'_{t,f}$ for a single time period t. In the benchmark and forecast routines, SS3 can calculate relF_f using $F'_{t,f}$ over a range of years, or the user can input custom rel$F$ values for benchmark and forecast in the forecast.ss file. Note that in a multi-season model setup, relF_f is implemented as rel$F_{s,f}$ where s is the season. These get multiplied by season duration as they are used.

In the benchmark section of the code, SS3 searches for an Fmult to achieve various management reference points (often referred to as benchmarks). In this search, SS3 calculates a benchmark F as $F'_{ben,f} = F\text{mult} * \text{rel}F_f$, then calculates equilibrium yield and spawning biomass per recruit (SPR). SS3 searches for the Fmult that satisfies the search conditions, first for user-specified SPR, then for user-specified spawning biomass at a management target (B_{TGT} or $F_{0.1}$), then for MSY. The resultant benchmark quantities are reported in the derived quantities, but Fmult and $F'_{ben,f}$ are only reported in the Forecast_report.sso file. SS3 stores the benchmark Fmult values so that user can invoke them for the forecast.

Annual F

The ann$F$ is a single annual value across all fleets and areas according to F_report_units, which is specified by users in the starter file. If there are many fleets, across several areas and with very different selectivity patterns, ann$F$ can have a complicated relationship to apical F. The F_report_units specification in the starter.ss file, see example line below, allows user to calculate it using F' directly, use exploitation rate, or be derived from Z-at-age.

Example F reporting unit specification in the starter.ss file:

```
5               # F_report_units:
                0 = skip;
                1 = exploitation(Bio);
                2 = exploitation(Num);
                3 = sum(Frates);
                4 = true F for range of ages;
                5 = unweighted avg. F for range of ages.
```

```
3 7             # min and max age over which average F will be calculated
```

For options 4 and 5 of F_report_units, the F is calculated as $Z - M$ where Z is calculated as $ln(N_{t+1,a+1}/N_{t,a})$, thus Z subsumes the effect of F.

The $\text{ann}F$ is calculated for each year of the estimated time series and of the forecast. Additionally, an $\text{ann}F$ is calculated in the benchmark calculations to provide equilibrium values that have the same units as $\text{ann}F$ from the time series. In versions previous to v.3.30.15, it was labeled inaccurately as Fstd in the output, not $\text{ann}F$. For example, in the Management Quantities section of derived quantities prior to v.3.30.15, there is a quantity labeled Fstd_Btgt. This is more accurately labeled as the annual F associated with the biomass target, ann_F_Btgt, in v.3.30.15.

Fstd

Fstd is a single annual value based on $\text{ann}F$ and the relationship to $\text{ann}F$ is specified by F_report_basis in the starter.ss file. The benchmark $\text{ann}F$ may be used to rescale the time series of $\text{ann}F$s to become a time series of standardized values representing the intensity of fishing, Fstd. The report basis is selected in the starter file as:

```
0               # F_report_basis:
                0 = raw F report;
                1 = F / F_{SPR};
                2 = F / F_{MSY};
                3 = F / F_{BTGT}.
```

For example, if user selects option 1, F / F_{SPR}, the time series of $\text{ann}F$ will be divided by each value by the $\text{ann}F$ calculated in benchmark.

Units for Stock Synthesis inputs related to F

Below is a list of items to consider in terms of units for F in SS3:

- If F_ballpark is specified in the control.ss file, its units are the same as ann$F$, so is not fleet-specific.

- F as parameter values has units of fleet-specific apical F'.

- In the forecast.ss file there is an option to input a vector of rel$F$ values. These are dimensionless and will be rescaled to sum to 1.0.

- In the forecast.ss file there is an option to specify an F scalar for the forecast. The units of F scalar are the same as the Fmult values calculated in benchmark. There are a full set of options for forecast F scalar that can be selected in the forecast file If the forecast F scalar is set as F_{SPR}, then SS3 will use SPR_Fmult calculated in benchmark and reported in Forecast-report.sso. If user selects the option to input an annual F scalar, option 5, then the value is input on a following line. Whichever method the user selects for forecast F scalar (Fmult), SS3 will start the forecast by creating a fleet-specific vector of apical F values from Fmult*relF_f.

- Also in the forecast.ss file, the last section of inputs allows for input of time and fleet specific apical $F'_{t,f}$ values that override the basic forecast F specification described above.

Guidance on Population Dynamics Modeling

Numerous assessment related topics have arisen among users. A collection of general information on population dynamics modeling is provided here. This information is relative to both users of SS3 and users of other stock assessment modeling platforms.

Data Weighting

In 2015 there was a CAPAM workshop dedicated to data weighting. Description of the workshop can be found on the CAPAM website. The presentations from the workshop are available through that website and many of them were included in a special issue of Fisheries Research.

Currently, there are three main methods for weighting length and data applied for U.S. West Coast assessments using Stock Synthesis.

1. McAllister - Ianelli: Effective sample size is calculated from fit of observed to expected length or age compositions. Tuning algorithm is intended to make the arithmetic mean of the input sample size equal to the harmonic mean of the effective sample size (McAllister and Ianelli, 1997).

2. Francis: Based on variability in the observed mean length or age by year, where the sample sizes are adjusted such that the fit of the expected mean length or age should fit within the uncertainty intervals at a rate which is consistent with variability expected based on the adjusted sample sizes (Method "TA1.8") (Francis and Hilborn, 2011)

3. Dirichlet-Multinomial: A new likelihood (as opposed to the standard multinomial) which includes an estimable parameter (theta) which scales the input sample size. In this case, the term "Effective sample size" has a different interpretation than in the McAllister-Ianelli approach (Thorson et al., 2017).

Applying the methods

McAllister-Ianelli
The "Length_Comp_Fit_Summary" and "Age_Comp_Fit_Summary" sections in the Report file include information on the harmonic mean of the effective sample size and arithmetic mean of the input sample size used in this tuning method. In the r4ss package, these tables are returned by the `SS_output` function as `$Length_comp_Eff_N_tuning_check` and `$Age_comp_Eff_N_tuning_check`.

A convenient way to process these values into the format required by the control file is to use the function:

```
SS_tune_comps(replist, option = "MI")
```

where the input "replist" is the object created by `SS_output`. This function will return a
table and also write a matching file called "suggested_tuning.ss" to the directory where the
model was run.

For version 3.30 models, the table created by `SS_tune_comps` can be pasted into the
bottom of the control file in the section labeled "Input variance adjustments", followed by
the terminator line which indicates the end of the section.

Also see the help page for the r4ss function `SS_varadjust` which can be used to
automatically write a new control file if you want to streamline the process of applying
multiple iterations of this tuning method.

If the tuning has been implemented, the green lines in the figure below would
approximately intersect at a point which is on the black 1-to-1 diagonal line in this figure
created by the r4ss function `SS_plots`.
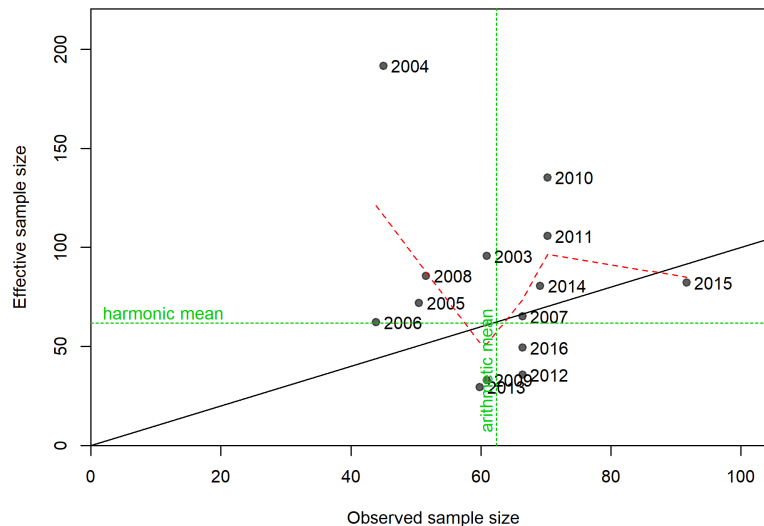
Figure 4: The relationship between the observed sample size (the input sample number)
versus the effective sample size where the effective sample size is the product of the input
sample size and the data weighting applied to the data set.

There are a couple of challenges posed by the McAllister-Ianelli data-weighting
approach:

1. Subjective choice of how many iterations to take to achieve adequate convergence.
   Often just one iteration is applied.

236

2. Takes time to implement so tuning is rarely repeated during retrospective or sensitivity analyses.

Francis

Implementation: recommended adjustments are calculated by the r4ss functions `SSMethod.TA1.8` and `SSMethod.Cond.TA1.8`. These functions are rarely used alone but are called by the `SS_plots` function when making plots like the one below. For version 3.30 models, the simplest way to get the adjustments in the format required by the control file is to use the `SS_tune_comps` function (described above under the McAllister-Ianelli method), but with a different option specified:

```
SS_tune_comps(replist, option = "Francis")
```

The figure below shows the estimated 95% intervals around the observed mean length by year based on the input sample size (thick lines) and the increase in that uncertainty which would occur if the sample sizes were adjusted according to the proposed multiplier.
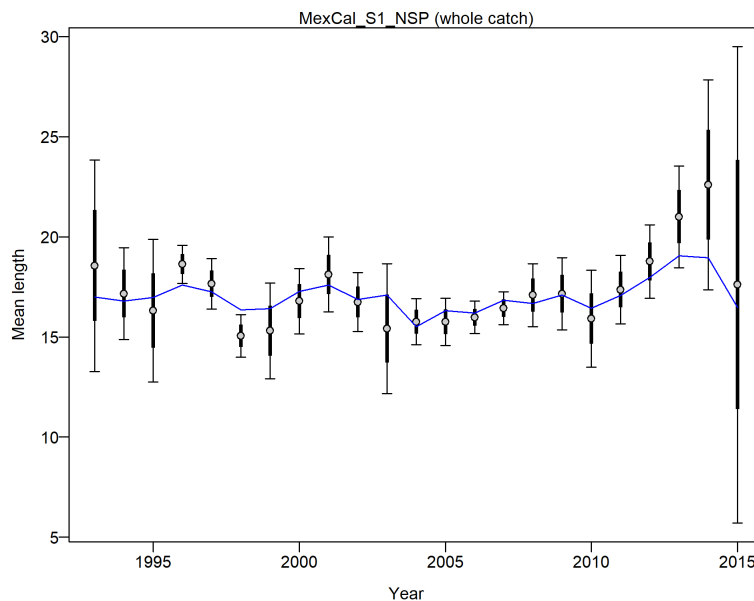
Figure 5: The mean length of the length samples for each year from the MexCal S1 NSP fleet with 95% confidence intervals based on current samples sizes using the Francis data weighting method (referred to as TA1.8). Thinner intervals with capped ends show result of further adjusting sample sizes based on the suggested multiplier, 0.2739, with 95% intervals for length data from the fleet. The blue line shows the expected variation of the mean length across years.

There are a several of challenges posed by the Francis data-weighting approach:

1. Subjective choice of how many iterations to take to achieve adequate convergence. Often just one iteration is applied.

2. Takes time to implement so tuning is rarely repeated during retrospective or sensitivity analyses.

3. Recommended adjustment can be sensitive to outliers (remove a few years of anomalous composition data can lead to large change in recommended adjustment).

Finally, in simulation work comparing both the Francis and McAllister and Ianelli data weighting approaches, indicate that the arithmetic averaging of effective samples sizes from the McAllister and Ianelli approach was inferior to other methods (Punt, 2016).

Dirichlet-multinomial

The Dirichlet-Multinomial should only be used when there is a substantive basis for setting an approximate value for input sample size. This is important because:

- The input-sample-size provided by the user is an upper bound on weighting for those data, such that a dummy value of 1 will cause those data to never be assigned a weight greater than 1.

- Changes in input sample size are not quite proportionally offset by changes in the estimated weighting parameter, such that using input-sample-size of 100 vs. 200 will result in small differences in estimated data weights (this property is fixed by the multivariate-Tweedie likelihood, where changes in input-sample-size are exactly offset by changes in the estimated weighting parameters).

To develop an input-sample-size, Jim Thorson recommends:

1. Using nonparametric (bootstrapping), design-based, or model-based estimators to identify the variance of expanded compositional data, and then deriving an input-sample-size from that;

2. Using a priori reasoning, e.g., about fishery compositions deserving a lower input-sample-size than a survey in cases when the input-sample-size for the survey is known, but that for the fishery is unknown.

3. If the proceeding options are not feasible, using accepted standards in that region and review context, i.e., assigning input-sample-size of arbitrary value X if that is what has always been done in that region/stock, there's no basis for changing it, and it is greater than or equal to the likely effective sample size;

Change the choice of likelihood and set parameter choices in the data file:

- In the specification of the length and/or age data, change "CompError" column in age and length comp specifications from 0 to 1 and "ParmSelect" from 0 to a sequence of numbers from 1 to N where N is the total number of combinations of fleet and age/length. Note that there cannot be any skipped values from 1 to N, otherwise the model will exit on error while reading the input files.

- Resulting input should look similar to:

```
#_mintail addtocomp combM+F CompressBins CompError ParmSelect minsamplesize
-1         0.001     0       0            1         1          0.001 #_fleet:1
-1         0.001     0       0            1         2          0.001 #_fleet:2
```

- The ParmSelect column can also have repeated values so that multiple fleets share the same log(Theta) parameter.

- If you have both length and age data, the ParmSelect should have separate numbers for each, e.g., 1 and 2 for the length comps and 3 and 4 for the age comps for the same two fleets.

Add parameter lines to the control file:

- Add as many parameter lines as the maximum numbers in the ParmSelect column. The new parameter lines go after the main selectivity parameters but before any time-varying selectivity parameters

- Given that the input-sample-size is fixed at a reasonable level, Jim Thorson recommends bounds of -5 to 5, with a starting value of 0 (which corresponds to a weight of about 50% of the input sample size). However, parameter estimates above 5 are associated with 99-100% weight with little information in the likelihood about the parameter value. Therefore, an upper bound of 5 may help identify cases that otherwise would have convergence issues. When the parameter hits this upper bound, consider either switching those data to a multinomial (which behavior the DM is asymptotically approaching) or using a Bayesian prior / likelihood penalty to ensure that the estimated parameter does not rest on this boundary.

- In consultation with Jim Thorson, Ian Taylor proposed a normal $N(0, 1.813)$ prior for the `ln(DM_parm)` parameters which to counteract the effect of the logistic transformation between this parameter and the data weighting. The 1.813 value was calculated as the standard deviation of the distribution of $log(\theta)$ values derived from starting with a uniform distribution on the weights, weight $= \theta/(1 + \theta) \sim U(0, 1)$, and solving for $log(\theta)$. This prior was applied to the 2020 Pacific Hake stock assessment (Grandin et al., 2020) and led to better MCMC convergence with relatively little impact on the maximum likelihood estimates.

- Example parameter lines are below (columns 8-14 not shown):

```
#_LO  HI   INIT  PRIOR PR_SD PR_type PHASE ... # Parm_name
-5    20   0     0     1.813 6       2     ... # ln(DM_theta)_1
-5    20   0     0     1.813 6       2     ... # ln(DM_theta)_2
```

- Reset any existing variance adjustments factors that might have been used for the McAllister-Ianelli or Francis tuning methods. In v.3.24 this means setting the values to 1, in v.3.30, you can simply delete or comment-out the rows with the adjustments.

The `SS_output` function in r4ss returns table like the following:

```
$Dirichlet_Multinomial_pars
                  Value Phase Min Max    Theta Theta/(1+Theta)
ln(DM_theta)_1 -0.164022     2  -5  20 0.8487233       0.4590862
ln(DM_theta)_2  2.246280     2  -5  20 9.4525070       0.9043292
```

The ratio shown in the final column is the estimated multiplier which can be compared to the sample size adjustment estimated in the other tuning methods above (the New_Var_adj column in the table produced by the `SS_tune_comps` function in r4ss).

If the reported $\theta/(1+\theta)$ ratio is close to 1.0, that indicates that the model is trying to tune the sample size as high as possible. In this case, the $ln(\theta)$ parameters should be fixed at a high value, like the upper bound of 20, which will result in 100% weight being applied to the input sample sizes. An alternative would be to manually change the input sample sizes to a higher value so that the estimated weighting will be less than 100

Note that a constant of integration was added to the Dirichlet-multinomial likelihood equation in v.3.30.17. This will change the likelihood value, but parameter estimates and expected values should remain the same as in previous versions of SS3.

Some challenges posed by the Dirichlet-multinomial data-weighting approach:

1. Does not allow weights above 100% (by design) so it is not yet clear how best to deal with the situation when the estimated weight is close to 100%.

2. Parameterization has potential to cause convergence issues or inefficient MCMC sampling when weights are close to 100% if no prior is applied as discussed above.

Recruitment Variability and Bias Correction

Recruitments are defined as lognormal deviates around a log-bias adjusted spawner-recruitment curve. The magnitude of the log-bias adjustment is calculated from the level of σ_R , which is the standard deviation of the recruitment deviations (in log-space). There are 5 segments of the time series in which to consider the effect of the log-bias adjustment: virgin; initial equilibrium; early data-poor period; data-rich period;

very-recent/forecast. The choice of break points between these segments need not correspond directly with the settings for the bias adjustment, although some alignment might be desired. Methot and Taylor (2011) provide more detailed discussion of the bias adjustment than what is provided below but do not address the separation of time periods into separate segments. The approach is illustrated with figures associated the assessment for darkblotched rockfish (Gertseva and Thorson, James T, 2013).
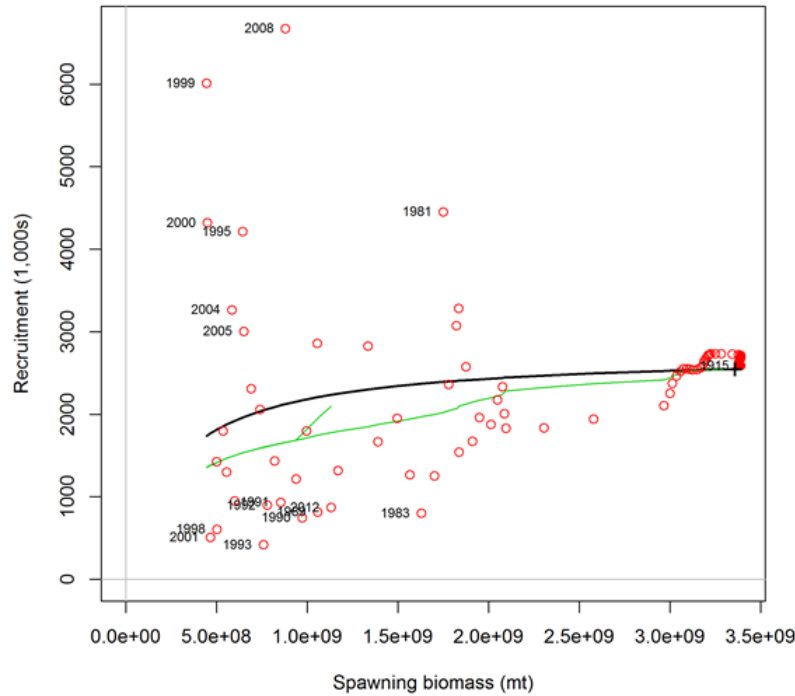

Figure 6: Spawner-recruitment relationship for dark-blotched rockfish (Gertseva and Thorson, James T, 2013). Red points represent estimated recruitments, the solid black line is the stock-recruit relationship and the green line represents the adjustment to this relationship after adjustment to account for the lognormal distribution associated with each year. The "+" symbol labeled 1915 near the right side represents both the virgin and initial equilibrium of the model. The numerous red points close to the initial conditions correspond to the early years of the model with low harvest rates.
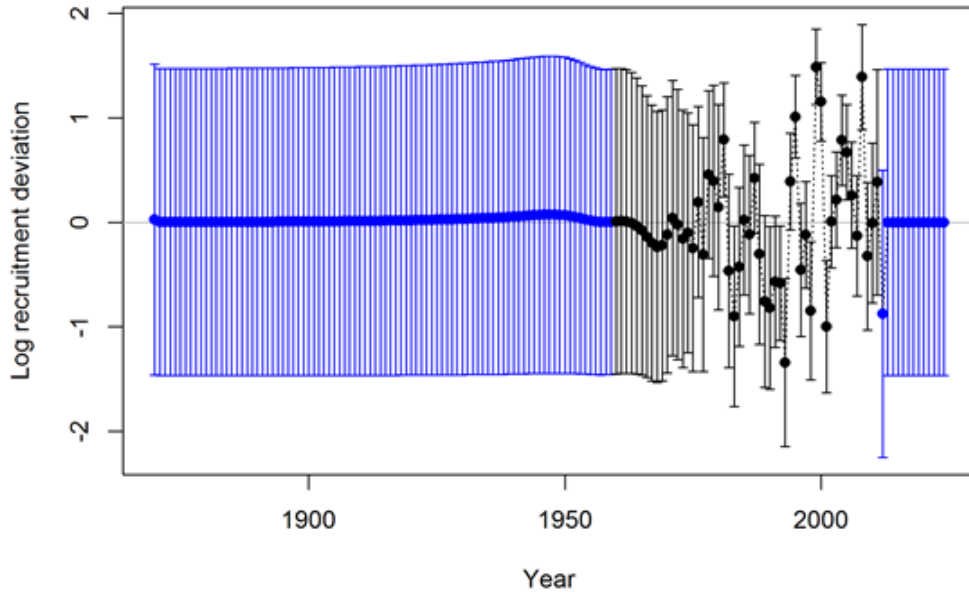
Figure 7: Time series of log recruitment deviations for darkblotched rockfish with 95% uncertainty intervals. The start year of the model is 1915, but recruitment deviations are estimated starting in 1870. The 45 deviation estimates for 1870–1914 inform the age structure used in the start year. The black color for the years 1960–2011 indicates the "main" recruitment deviation vector, while the blue color for the years 1870–1959 and 2012–2024 indicates the "early" and "late/forecast" recruitment deviation vectors, respectively.

Virgin Biomass

The R0 level of recruitment is a parameter of the spawner-recruitment curve. This recruitment and the corresponding spawning biomass are expected to represent the long-term arithmetic mean.

Initial Equilibrium

The level of recruitment is typically maintained at the R0 level even though the initial equilibrium catch will reduce the spawning biomass below the virgin level. If steepness is moderately low or the initial F is high, then the lack of response in recruitment level may appear paradoxical. The logic is that building in the spawner-recruitment response to initial F would significantly complicate the calculations and would imply that the initial equilibrium catch level had been going on for multiple generations. If the lack of response is considered to be problematic in a particular application, then start the model at an earlier year and with a lower initial equilibrium catch so that the dynamics of the spawner-recruitment response get captured in the early period, rather than getting lost in the initial equilibrium.

242

Early Data-Poor Period

This is the early part of the time series where the only data typically are landed catch. There are no data to inform the model about the specific year-to-year fluctuations in recruitment, although the ending years of this period will begin to be influenced by the data. The "early time period" is not a formal concept. It is up to the user to decide whether to start estimating recruitment deviations beginning with the first year of the model, or to delay such estimation until the data become more informative. Modeling recruitment deviations in this period may lead to a more realistic portrayal of the uncertainty in depletion, but can also lead to spurious patterns in estimated recruitments that may be driven by the fit to index data or other sources that would not be expected to have accurate information on recruitment.

- Option A: Do not estimate recruitment deviations during this early period. During years prior to the first year of recruitment deviations, the model will set the recruitment equal to the level of the spawner-recruitment curve. Thus, it is a mean-based level of recruitment. Because these annual parameters are fixed to the level of the spawner-recruitment curve, they have no additional uncertainty and make no contribution to the variance of the model. This approach may produce relatively large, or small, magnitude deviations at the very beginning of the subsequent period, as the model "catches up" to any slight signal that could not be captured through estimated deviations in the early data-poor period. There may be some effect on the estimate of R0 as a result of lack of model flexibility in balancing early period removals with signal in the early portion of the data-rich period.

- Option B: Estimate recruitment deviations for all the early years. Each of these recruitment deviations is now a deviation parameter so will have a variance that contributes to the total model variance. The estimated standard deviation of each of these deviation parameters should be similar to σ_R because σ_R is the only constraint on these parameters (however, the last few in the sequence will begin to feel the effect of the data so may have lower standard deviations).

Data-Rich Period

Here the length and or age data inform the model on the year-to-year level of recruitment. These fluctuations in recruitment are assumed to have a lognormal distribution around the log-bias adjusted spawner-recruitment curve. The level of σ_R input to the model should match this level of fluctuation to a reasonable degree. Because the recruitments are lognormal, they produce a mean biomass level that is comparable to the virgin biomass and thus the depletion level can be calculated without bias. However, if the early period has recruitment deviations estimated by maximum posterior density, then the depletion levels during the early part of the data-rich period may have some lingering effect of negative bias during the early time period. The level of σ_R should be at least as large as the level of variability in these estimated recruitments. If too high a level of σ_R is used, then a bias can occur in the estimate of spawner-recruitment steepness, which determines

the trend in recruitment. This occurs when the early recruitments are taken directly from the spawner- recruitment curve, so are mean unbiased, then the later recruitments are estimated as deviations from the log-bias adjusted curve. If σ_R is too large, then the bias-adjustment is too large, and the model may compensate by increasing steepness to keep the mean level of recent recruitments at the correct level.

Recent Years/Forecast

Here the situation is very similar to the early time period in that there are no data to inform the model about the year-to-year pattern in recruitment fluctuations so all deviations will be pulled to a zero level in the maximum posterior density. The structure of SS3 creates no sharp dividing line between the estimation period and the forecast period. In many cases one or more recruitments at the end of the time series will lack appreciable signal in the data and should therefore be treated as forecast recruit deviations. To the degree that some variability is observed in these recruitments, partial or full bias correction may be desirable for these deviations separate from the purely forecast deviations, there is therefore an additional control for the level of bias correction applied to forecast deviations occurring prior to end year $+$ 1.
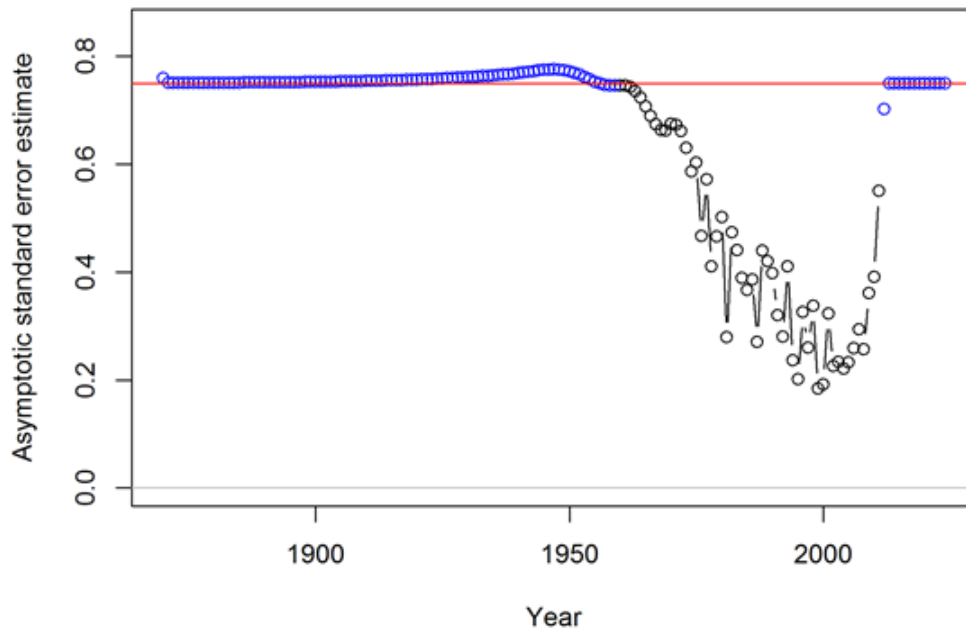

Figure 8: Timeseries of standard error estimates for the log recruitment deviations for darkblotched rockfish with 95% uncertainty intervals. As in Figure 7, the black color indicates the main recruitment period. This period with lower standard error is associated with higher variability among deviations (Figure 7). The red line at 0.75 indicates the σ_R value in this model.
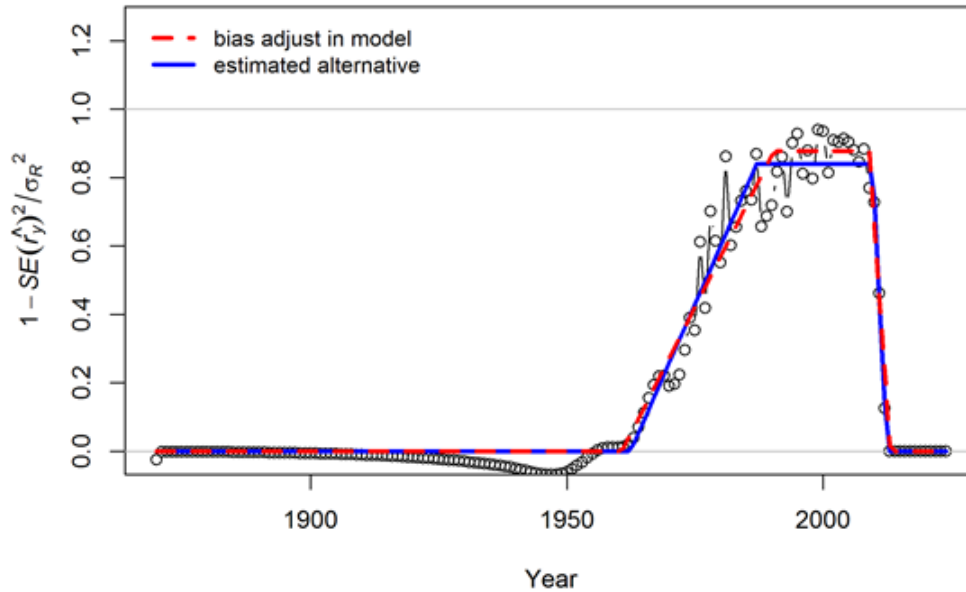
244

Figure 9: Transformation of the standard error estimates (shown in Figure 8) for darkblotched rockfish following the approach suggested by Methot and Taylor (2011). These values were used to set the 5 values controlling the degree of bias adjustment (as a fraction of $\sigma_R/2$) to account for differences in the mean and median of the lognormal distribution from which the recruitment deviations are drawn. The red line indicates a bias adjustment of 0 up to the 1960.75, ramping up to a maximum adjustment level of 0.877 for the period 1990.4–2008.98,and reducing back to 0 starting in 2013.08. Note that these values controlling the bias adjustment need not be integer year values. Also the break points in the bias adjustment function need not match the break points between early, main, and late/forecast recruitment deviation vectors (indicated by blue and black colors in Figures 7 and 8). The blue line indicates a functional form that minimizes the sum of squared differences between the bias adjustment function and the transformed standard error values. The subtle differences between red and blue lines are unlikely to have any appreciable effect on the model results.
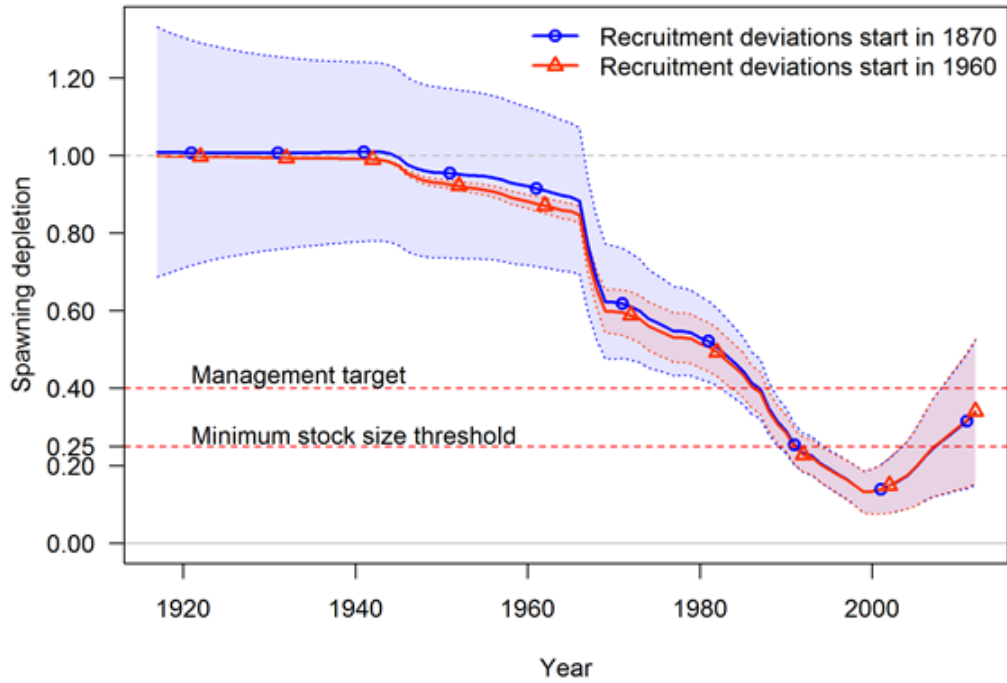
Figure 10: Comparison of time series of spawning depletion for darkblotched rockfish models with early recruitment deviations (starting in 1870) and without early deviations (only main recruitment deviations starting in 1960). The point estimates are similar, but the 95% uncertainty intervals are substantially different. With no recruitment deviations for the early period, the estimates of spawning depletion in the early years are very precise and uncertainty increases as the stock moves into the data rich period. In contrast, the addition of the early recruitment deviations results in a large uncertainty in spawning depletion for the early years and an increase in precision as the stock moves into the data rich period. In this application, the uncertainty associated with the recent years is independent of the assumptions about early recruitments.

Issues with Including Environmental Effects

The expected level of recruitment is a function of spawning biomass, an environmental time series, and a log-bias adjustment.

$$E(Recruitment) = f(SpBio) * exp(\beta * envdata) * exp(-0.5 * \sigma_R^2) \qquad (61)$$

σ_R is the variability of the deviations, so it is in addition to the variance "created" by the environmental effect. So, as more of the total recruitment variability is explained by the environmental effect, the residual σ_R should be decreased. The model does not do this automatically.

246

The environmental effect is inherently lognormal. So when an environmental effect is included in the model, the arithmetic mean recruitment level will be increased above the level predicted by f(SpBio) alone. The consequences of this have not yet been thoroughly investigated, but there probably should be another bias correction based on the variability of the environmental data as scaled by the estimated linkage parameter, β. It is also problematic that the environmental effect time series used as input is assumed to be measured without error.

The preferred approach to including environmental effects on recruitment is not to use the environmental effect in the direct calculation of the expected level of recruitment. Instead, the environmental data would be used as if it was a survey observation of the recruitment deviation. This approach is similar to using the environmental index as if it was a survey of age 0 recruitment abundance because by focusing on the fit to the deviations it removes the effect of SpBio on recruitment. In this alternative, the σ_R would not be changed by the environmental data; instead the environmental data would be used to explain some of the total variability represented by σ_R. This approach may also allow greater uncertainty in forecasts, as the variability in projected recruitments would reflect both the uncertainty in the environmental observations themselves and the model fit to these observations.

Initial Age Composition

If the first year with recruitment deviations is set less than the start year of the model, then these early deviations will modify the initial age composition. The amount of information on historical recruitment variability certainly will degrade as the model attempts to estimate deviations for older age groups in the initial equilibrium. So the degree of bias correction is reduced linearly in proportion to age so that the correction disappears when maximum age is reached. The initial age composition approach normally produces a result that is indistinguishable from a configuration that starts earlier in the time series and estimates a longer time series of recruitments. However, because the initial equilibrium is calculated from a recruitment level unaffected by spawner-recruitment steepness and initial age composition adjustments are applied after the initial equilibrium is calculated, it is possible that the initial age composition approach will produce a slightly different result than if the time series was started earlier and the deviations were being applied to the recruitment levels predicted from the spawner-recruitment curve.

Bibliography

Francis, R.C. and Hilborn, R. 2011. Data weighting in statistical fisheries stock assessment models. Canadian Journal of Fisheries and Aquatic Sciences 68(6): 1124–1138. doi:10.1139/f2011-025.

Gertseva, V. and Thorson, James T 2013. Status of the darkblotched rockfish resource off the continental U.S. Pacific Coast in 2013. Technical report, Pacific Fishery Management Council, 7700 Ambassador Place NE, Suite 200, Portland, OR 97220.

Grandin, C.J., Johnson, K.F., Edwards, A.M., and Berger, A.M. 2020. Status of Pacific Hake (whiting) stock in the U.S. and Canadian waters in 2020. Technical report, Joint Technical Committe of the Pacific Hake/Whiting Agreement Between the Governments of the United States and Canada, National Marine Fisheries Service and Fisheries and Ocean Canada.

Gulland, J. 1987. Natural mortality and size. Marine Ecology Progress Series 39(2): 197–199.

Johnson, K.F., Councill, E., Thorson, J.T., Brooks, E., Methot, R.D., and Punt, A.E. 2016. Can autocorrelated recruitment be estimated using integrated assessment models and how does it affect population forecasts? Fisheries Research 183: 222–232. doi:10.1016/j.fishres.2016.06.004.

Lee, H., Piner, K.R., Taylor, I.G., and Kitakado, T. 2019. On the use of conditional age at length data as a likelihood component in integrated population dynamics models. Fisheries Research 216: 204–211. doi:10.1016/j.fishres.2019.04.007.

Lee, H.H., Thomas, L.R., Piner, K.R., and Maunder, M.N. 2017. Effects of age-based movement on the estimation of growth assuming random-at-age or random-at-length data. Journal of Fish Biology 90(1): 222–235. doi:10.1111/jfb.13177.

Lehodey, P., Senina, I., and Murtugudde, R. 2008. A spatial ecosystem and populations dynamics model (seapodym)–modeling of tuna and tuna-like populations. Progress in Oceanography 78(4): 304–318.

Lorenzen, K. 1996. The relationship between body weight and natural mortality in juvenile and adult fish: a comparison of natural ecosystems and aquaculture. Journal of Fish Biology 49(4): 627–642.

Lorenzen, K. 2000. Allometry of natural mortality as a basis for assessing optimal release size in fish-stocking programmes. Canadian Journal of Fisheries and Aquatic Sciences 57(12): 2374–2381.

Maunder, M.N. 2011. Proposed formulation for age-specific patterns in natural mortality. In Estimating natural mortality in stock assessment applications, edited by J. Brodziak,

J. Ianelli, K. Lorenzen, and J. Methot R.D., U.S. Department of Commerce, NOAA Tech. Memo, Silver Spring, Maryland, USA, p. 38.

Maunder, M.N. and Deriso, R.B. 2003. Estimation of recruitment in catch-at-age models. Canadian Journal of Fisheries and Aquatic Sciences 60(10): 1204–1216. doi:10.1139/f03-104.

Maunder, M.N., Aires-da Silva, A., Deriso, R., Schaefer, K., and Fuller, D. 2010. Preliminary estimation of age-and sex-specific natural mortality of bigeye tuna in the eastern pacific ocean by applying a cohort analysis with auxiliary information to tagging data. Inter-Amer. Trop. Tuna Comm., Stock Assessment Report 10: 253–278.

Maunder, M.N., Deriso, R.B., Schaefer, K.M., Fuller, D.W., Aires-da Silva, A.M., Minte-Vera, C.V., and Campana, S.E. 2018. The growth cessation model: a growth model for species showing a near cessation in growth with application to bigeye tuna (Thunnus obesus). Marine Biology 165(4). doi:10.1007/s00227-018-3336-9.

McAllister, M.K. and Ianelli, J.N. 1997. Bayesian stock assessment using catch-age data and the sampling - importance resampling algorithm. Canadian Journal of Fisheries and Aquatic Sciences 54: 284–300.

Methot, R.D. and Taylor, I.G. 2011. Adjusting for bias due to variability of estimated recruitments in fishery assessment models. Canadian Journal of Fisheries and Aquatic Sciences 68(10): 1744–1760. doi:10.1139/f2011-092.

Methot, R.D. and Wetzel, C.R. 2013. Stock synthesis: A biological and statistical framework for fish stock assessment and fishery management. Fisheries Research 142: 86–99. doi:10.1016/j.fishres.2012.10.012.

Monnahan, C.C., Branch, T.A., Thorson, J.T., Stewart, I.J., and Szuwalski, C.S. 2019. Overcoming long bayesian run times in integrated fisheries stock assessments. ICES Journal of Marine Science 76(6): 1477–1488. doi:10.1093/icesjms/fsz059.

Piner, K.R., Lee, H.H., and Maunder, M.N. 2016. Evaluation of using random-at-length observations and an equilibrium approximation of the population age structure in fitting the von Bertalanffy growth function. Fisheries Research 180: 128–137. doi:10.1016/j.fishres.2015.05.024.

Punt, A.E. 2016. Some insights into data weighting in integrated stock assessments. Fisheries Research 192: 52–65. doi:10.1016/j.fishres.2015.12.006.

Punt, A.E. and Cope, J.M. 2019. Extending integrated stock assessment models to use non-depensatory three-parameter stock-recruitment relationships. Fisheries Research 217: 46–57. doi:10.1016/j.fishres.2017.07.007.

Richards, F. 1959. A flexible growth function for empirical use. Journal of Experimental Botany 10(29): 290–300.

Schnute, J. 1981. A versatile growth model with statistically stable parameters. Canadian Journal of Fisheries and Aquatic Science 38: 1128–1140.

Taylor, I.G., Gertseva, V., Methot, R.D., and Maunder, M.N. 2013. A stock-recruitment relationship based on pre-recruit survival, illustrated with application to spiny dogfish shark. Fisheries Research 142: 15–21. doi:10.1016/j.fishres.2012.04.018.

Then, A.Y., Hoenig, J.M., Hall, N.G., Hewitt, D.A., and editor: Ernesto Jardim, H. 2015. Evaluating the predictive performance of empirical estimators of natural mortality rate using information on over 200 fish species. ICES Journal of Marine Science 72(1): 82–92.

Thompson, G.G. 1994. Confounding of Gear Selectivity and the Natural Mortality Rate in Cases where the Former is a Nonmonotone Function of Age. Canadian Journal of Fisheries and Aquatic Sciences 51(12): 2654–2664. doi:10.1139/f94-265.

Thorson, J.T., Johnson, K.F., Methot, R.D., and Taylor, I.G. 2017. Model-based estimates of effective sample size in stock assessment models using the Dirichlet-multinomial distribution. Fisheries Research 192: 84–93. doi:10.1016/j.fishres.2016.06.005.

Xu, H., Thorson, J.T., Methot, R.D., and Taylor, I.G. 2019. A new semi-parametric method for autocorrelated age- and time-varying selectivity in age-structured assessment models. Canadian Journal of Fisheries and Aquatic Sciences 76(2): 268–285. doi:10.1139/cjfas-2017-0446.